



DESENVOLVIMENTO DE UMA FERRAMENTA COMPUTACIONAL PARA MODELAGEM E SIMULAÇÃO DE CIRCUITOS ELETRÔNICOS DIGITAIS

Pedro Cezar Souza de Lucena

Projeto de Graduação apresentado ao Curso de Engenharia Elétrica da Escola Politécnica, Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Engenheiro.

Orientador: Antonio Lopes de Souza, Ph.D.

RIO DE JANEIRO

NOVEMBRO DE 2013

**DESENVOLVIMENTO DE UMA FERRAMENTA COMPUTACIONAL PARA
MODELAGEM E SIMULAÇÃO DE CIRCUITOS ELETRÔNICOS DIGITAIS**

Pedro Cezar Souza de Lucena

PROJETO DE GRADUAÇÃO SUBMETIDO AO CORPO DOCENTE DO CURSO DE
ENGENHARIA ELÉTRICA DA ESCOLA POLITÉCNICA DA UNIVERSIDADE
FEDERAL DO RIO DE JANEIRO, COMO PARTE DOS REQUISITOS NECESSÁRIOS
PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO ELETRICISTA.

Aprovada por:

Professor Antonio Lopes de Souza, Ph. D.
(orientador)

Professor Sergio Sami Hazan, Ph. D.

Professor Oumar Diene, D. Sc.

RIO DE JANEIRO, RJ - BRASIL

NOVEMBRO DE 2013

Souza de Lucena, Pedro Cezar

Desenvolvimento de uma Ferramenta Computacional para Modelagem e Simulação de Circuitos Eletrônicos Digitais / Pedro Cezar Souza de Lucena, Rio de Janeiro: UFRJ/Escola Politécnica/ Curso de Engenharia Elétrica, 2013.

vii, 146 p.: Il.; 29,7 cm.

Orientador: Antonio Lopes de Souza, Ph.D.

Projeto de Graduação – UFRJ/ Escola Politécnica/ Curso de Engenharia Elétrica, 2013.

Referências Bibliográficas: p. 136-136.

1.Eletrônica Digital. I.Souza, Antonio Lopes de. II.Universidade Federal do Rio de Janeiro, Escola Politécnica, Curso de Engenharia Elétrica. III.Título.

AGRADECIMENTOS

Agradeço ao meu pai Aurimendes Pereira de Lucena e minha mãe Maria Risonilda Souza de Lucena por todo o apoio, incentivo, dedicação, suporte e investimento que possibilitou o meu ingresso à Universidade Federal do Rio de Janeiro e a conclusão do curso de Engenharia Elétrica.

Agradeço à minha irmã Mariana de Fátima Souza de Lucena por todo o suporte e incentivo durante todos os anos de faculdade.

Agradeço a todos os professores do Departamento de Engenharia Elétrica pelo tempo dedicado e conhecimento transmitido.

Agradeço a todos os amigos da graduação pela ajuda e incentivo nos estudos.

Agradeço ao meu professor orientador Antonio Lopes de Souza que desde o início de minha vida acadêmica sempre me apoiou, incentivou e ajudou em minha formação. Agradeço também pela oportunidade de estagiar no Laboratório de Novas Tecnologias para o Ensino da Engenharia. Agradeço também pela dedicação em supervisionar meu projeto de graduação.

Resumo

A utilização de simuladores digitais para o ensino de engenharia é uma prática muito utilizada, pois permite ao estudante acesso rápido a resultados que só poderiam ser obtidos depois de complicadas montagens e análises utilizando ferramentas complexas, ou mesmo que não poderiam ser obtidas pela dificuldade de acesso a algum componente da montagem. A utilização de simuladores também permite maior fixação do conteúdo apresentado em sala de aula, pois o estudante aplica o que lhe foi ensinado ao realizar as montagens a serem simuladas. Há também o incentivo a explorar novas montagem e consequentemente desenvolver novos projetos que não são abordados em sala de aula e avançar no assunto muito além do proposto pelo professor.

O objetivo deste trabalho é apresentar e detalhar uma ferramenta computacional desenvolvida para a modelagem e simulação de circuitos eletrônicos digitais. Essa ferramenta é flexível e possibilita a simulação de qualquer circuito lógico criado por seu usuário, sendo limitada apenas pela disponibilidade dos componentes eletrônicos presentes na biblioteca. A ferramenta possui uma interface gráfica completa e amigável ao usuário, mantendo a informação e operação do circuito criado acessível em poucos passos.

A ferramenta computacional foi criada utilizando a linguagem de programação *Action Script 3.0*, linguagem orientada a objeto que permite criar versões do programa para diferentes sistemas operacionais sem a necessidade de grandes edições no código. Essa linguagem também permite a manipulação de elementos gráficos, assim toda a interface gráfica foi desenvolvida utilizando essa mesma linguagem.

O código fonte da ferramenta computacional está estruturado de maneira a permitir a inserção de novas funções, podendo assim ampliar o campo de atuação da ferramenta.

Sumário

Agradecimentos.....	iv
Resumo	v
Sumário	vi
1. Introdução	1
1.1 Objetivo	3
1.2 Descrição	3
2. Eletrônica Digital.....	4
2.1 Álgebra de Boole.....	4
2.2 Circuitos Eletrônicos Digitais	7
3. Análise do Circuito	10
3.1 Circuito Digital.....	10
3.2 Tradução Computacional.....	13
4. Descrição do Programa	16
4.1 Interface Gráfica.....	16
4.1.1 Barra de Ferramentas	17
4.1.1.1 INICIAR/CANCELAR	17
4.1.1.2 MAIS/MOSTRADOR/MENOS	18
4.1.1.3 DELETAR.....	18
4.1.1.4 SALVAR.....	18
4.1.1.5 ABRIR.....	19
4.1.1.6 IMPRIMIR	19
4.1.1.7 MINIMIZAR	19
4.1.1.8 MACROS	19
4.1.1.9 INTERPRETAR	20
4.1.1.10 AMPLIAR	20
4.1.1.11 REDUZIR.....	20
4.1.1.12 ARRASTAR.....	21
4.1.1.13 APAGAR.....	21
4.1.1.14 TRACE.....	21
4.1.1.15 PROBE	21
4.1.1.16 DUPLICAR	22
4.1.1.17 ATALHOS	22

4.1.1.18	ROTAÇÃO ANTI-HORÁRIA.....	22
4.1.1.19	ROTAÇÃO HORÁRIA	22
4.1.1.20	GRÁFICOS	23
4.1.1.21	DETALHES.....	23
4.2	Container de Portas.....	24
4.3	Barra de Informações.....	26
4.4	Área de Trabalho	27
4.5	Funções	28
4.5.1	Adição de Portas Lógicas	28
4.5.2	Criação de Fios	30
4.5.3	Simulação	35
4.5.4	Trace.....	39
4.5.5	Probe	41
4.5.6	Seleção.....	42
4.5.7	Arrasto	46
4.5.8	Ampliar / Reduzir	63
4.5.9	Eliminação	65
4.5.10	Duplicação	66
4.5.11	Rotação.....	68
4.5.12	Deletar	72
4.5.13	Salvar.....	73
4.5.14	Abrir	75
4.6	Editores	77
4.6.1	Editor de Propriedades	77
4.6.2	Editor de Minimização	90
4.6.3	Editor de Interpretação de Funções.....	99
4.6.4	Editor de Macros.....	108
4.6.5	Editor de Gráficos	115
4.6.6	Editor de Impressão	126
4.6.7	Editor de Atalhos	129
5.	Conclusões.....	134
6.	Bibliografia	136
	ANEXO 1 – Algoritmo de Quine McCluskey	137
	APÊNDICE A -	146

Capítulo 1

Introdução

Os programas de simulação computacional têm sido utilizados nos mais variados campos do conhecimento. Na área das engenharias estes desempenham papel fundamental na realização de estudos que precedem as montagens reais. A utilização de simuladores nesses estudos permite a criação e análise de quantas situações e montagens sejam necessárias afim de entender completamente e otimizar o funcionamento da montagem final. A vantagem da utilização de simuladores é que muitas das vezes construir e testar todas as configurações da montagem em estudo pode demandar uma grande quantidade de tempo e capital, ou até exista a impossibilidade de realizar essas montagens. Por outro lado um estudo analítico do funcionamento de todas as montagens pode ser muito complicado, assim a utilização de simuladores permite que o estudo seja realizado com economia de tempo e capital, e que se possa construir a montagem final de maneira segura e otimizada.

Dentre os simuladores mais populares está a ferramenta Simulink do programa Matlab, propriedade de Mathworks, Inc, um simulador de sistemas dinâmicos, que possui uma série de funções que permitem a comunicação com circuitos eletrônicos reais ligados ao computador. Isto torna possível a realização de análises bem como a atuação sobre o sistema estudado. Outro simulador conhecido é o programa ANAREDE, propriedade da Eletrobrás, que é uma ferramenta capaz de realizar análises e simulações do funcionamento de Sistemas Elétricos de Potência.

O simulador de circuitos eletrônicos Circuit Maker, é uma ferramenta computacional com a qual é possível desenhar circuitos, simulá-los e realizar estudos sobre o seu funcionamento. É possível estudar tanto circuitos eletrônicos analógicos quanto digitais, porém este programa teve seu desenvolvimento descontinuado e a versão mais atual data do ano 2000. Existe uma grande quantidade de programas desenvolvidos por grandes empresas que também permitem criar e analisar circuitos eletrônicos analógicos e digitais, porém apenas versões reduzidas são disponibilizadas de forma gratuita para estudantes limitando sua utilidade. Para ter acesso a

versão completa desses programas é necessário, via de regra, adquirir licenças, o que em sua maioria representam um gasto muito elevado.

Desse modo foi proposto o desenvolvimento de um programa computacional que permite a modelagem, simulação e análise de circuitos eletrônicos digitais e que será disponibilizado de forma gratuita. Esse programa apresenta um conjunto de ferramentas que podem ser utilizadas para auxiliar na criação desses circuitos e em sua análise. O programa foi desenvolvido com ferramentas de programação modernas que permitem que este seja executado em sistemas operacionais atuais, tirando proveito de toda a capacidade computacional que estes sistemas podem oferecer. O programa foi desenvolvido também com foco em disponibilizá-lo em outras plataformas atuais como smartphones, tablets, navegadores sem a necessidade de instalação.

O presente trabalho apresenta o programa computacional desenvolvido, detalhando sua interface gráfica e o funcionamento de cada função e de cada editor presente no programa. Algumas explicações teóricas serão feitas para ambientar o leitor que não possua conhecimentos prévios sobre a eletrônica digital. Além disso será apresentado como foi abordada a simulação de circuitos reais em um meio computacional, detalhando os aspectos que foram levados em consideração e os que foram desconsiderados nesse processo.

1.1. Objetivo

O objetivo deste trabalho é apresentar e detalhar uma ferramenta computacional desenvolvida para a modelagem e simulação de circuitos eletrônicos digitais. Essa ferramenta é flexível e possibilita a modelagem e simulação de circuitos lógicos combinacionais e sequenciais, sendo limitada apenas pela disponibilidade de modelos dos componentes eletrônicos existentes na biblioteca do programa desenvolvido. A ferramenta possui uma interface gráfica completa e amigável ao usuário, mantendo a informação e operação do circuito criado acessível em poucos passos.

1.2. Descrição

Inicialmente será realizada uma revisão teórica básica da eletrônica digital, abordando temas como a Álgebra de Boole, os tipos de portas lógicas e os Circuitos Eletrônicos Digitais.

Posteriormente será realizada uma análise da dinâmica de funcionamento de um circuito eletrônico digital, mostrando como os componentes do circuito se comunicam e como o circuito se comporta durante uma análise de seu funcionamento.

Após a análise do funcionamento do circuito eletrônico digital será mostrado como foi feita a tradução de um circuito real para o circuito simulado no programa. Será mostrado ainda o modo como foram traduzidas as funções das portas lógicas presentes no programa e as situações de curto circuito.

Será mostrada a interface gráfica do programa e serão apresentadas todas as funções e editores do programa. As funções e editores terão o seu funcionamento completamente detalhado.

O Algoritmo de Quine - McCluskey, aplicado para minimização de funções lógicas e usado na implementação das rotinas de minimização de funções do programa será apresentado e detalhado no Anexo 1.

Capítulo 2

Eletrônica Digital

2.1. Álgebra de Boole

A álgebra de Boole é um sistema matemático composto por operadores, regras, postulados e teoremas [1]. Ela usa funções e variáveis que podem assumir apenas um de dois valores, zero ou um. As operações lógicas fundamentais da Álgebra de Boole são: soma lógica, produto lógico e negação lógica [1].

A operação de soma lógica é definida como sendo aquela em que a saída assume o estado lógico um quando pelo menos uma das variáveis de entrada apresenta estado lógico um. Assim a única possibilidade da saída apresentar como resposta estado lógico zero é se nenhuma de suas variáveis de entrada apresentar estado lógico um. A operação de soma lógica é representada pelo operador OR, que é simbolizado pelo sinal “+” que deve ser inserido entre as variáveis de entrada da função, “ $A+B$ ”. A Figura 2.1 apresenta a Tabela Verdade desta função para duas variáveis de entrada e o símbolo da porta lógica que representa essa função.

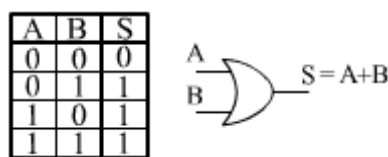


Figura 2.1 – Tabela Verdade e símbolo da porta lógica da função soma lógica.

A operação de produto lógico é definida como sendo aquela em que a saída assume o estado lógico um somente quando todas as variáveis de entrada apresentam estado lógico um. Caso uma das variáveis de entrada apresente estado lógico zero, a saída assume estado lógico zero. A operação de produto lógico é representada pelo operador AND, que é simbolizado pelo sinal “.” que deve ser inserido entre as variáveis de entrada da função, “ $A.B$ ”. A Figura 2.2 apresenta a Tabela Verdade desta função para duas variáveis de entrada e o símbolo da porta lógica que representa essa função.

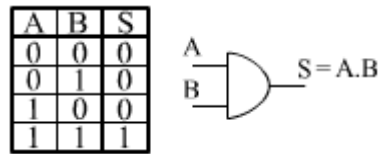


Figura 2.2 – Tabela Verdade e símbolo da porta lógica da função produto lógico.

A operação de negação é definida como sendo aquela em que a saída assume o estado lógico complementar ao da variável de entrada. A operação de negação também pode ser chamada de complementação e é representada pelo operador NOT, que é simbolizado por um traço que é inserido sobre a função ou variável de entrada. Em alguns casos o operador é simbolizado por um ponto de exclamação que é inserido imediatamente a esquerda da variável de entrada, “!A”. A Figura 2.3 apresenta a Tabela Verdade desta função, e o símbolo da porta lógica que representa essa função.

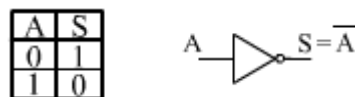


Figura 2.3 – Tabela Verdade e símbolo da porta lógica da função complementação.

As operações lógicas descritas podem ser referenciadas pelo nome de seus operadores, assim as operações de produto lógico, soma lógica e negação lógica podem ser referenciadas respectivamente como AND, OR e NOT. A partir dessas operações básicas é possível desenvolver operações lógicas secundárias que também são referenciadas pelo nome de seus operadores e essas são: NAND, NOR, EXOR e EXNOR.

A operação lógica NAND é definida como sendo aquela em que a saída assume o estado lógico zero somente quando todas as variáveis de entrada apresentam estados lógicos um. Ela pode ser interpretada como uma operação de complementação aplicada a uma função de produto lógico. A Figura 2.4 apresenta a Tabela Verdade desta função e o símbolo da porta lógica que a representa.

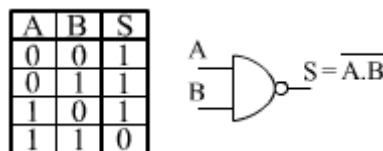


Figura 2.4 – Tabela Verdade e símbolo da porta lógica da operação lógica NAND.

A operação lógica NOR é definida como sendo aquela em que a saída assume o estado lógico um somente quando todas as variáveis de entrada apresentam estados lógicos zero.

Essa operação pode ser interpretada como uma operação de complementação aplicada a uma função de soma lógica. A Figura 2.5 apresenta a Tabela Verdade desta função e o símbolo da porta lógica que a representa.

A	B	S
0	0	1
0	1	0
1	0	0
1	1	0

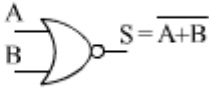


Figura 2.5 – Tabela Verdade e símbolo da porta lógica da operação lógica NOR.

A operação lógica EXOR é definida como sendo aquela em que a saída assume o estado lógico um somente quando existe número ímpar de variáveis de entrada que apresentem estado lógico um. Essa operação é simbolizada pelo operador “ \oplus ” que deve ser inserido entre as variáveis de entrada, “ $A \oplus B$ ”. A Figura 2.6 apresenta a Tabela Verdade desta função e o símbolo da porta lógica que representa essa função.

A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

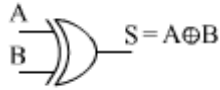


Figura 2.6 – Tabela Verdade e símbolo da porta lógica da operação lógica EXOR.

A Álgebra de Boole também pode ser chamada de álgebra dos circuitos lógicos, uma vez que a partir dela é possível compreender o funcionamento das portas lógicas utilizadas em um circuito lógico. Os símbolos que representam essas portas lógicas foram apresentados nas Figuras 2.1, 2.2, 2.3, 2.4, 2.5 e 2.6, e essas portas lógicas são referenciadas pelo nome de seus operadores, assim são chamadas respectivamente de porta lógica OR, AND, NOT, NAND, NOR e EXOR.

2.2. Circuitos Eletrônicos Digitais

Os circuitos eletrônicos digitais fazem parte de um ramo da eletrônica em que as informações processadas por seus componentes são apresentadas na forma de funções discretas no tempo, assumindo valores padronizados que representam o estado lógico um ou estado lógico zero [1].

Os circuitos eletrônicos digitais modernos são construídos utilizando Circuitos Integrados. Um Circuito Integrado possui em seu interior um ou mais circuitos eletrônicos compostos principalmente por transistores, diodos, resistores e capacitores [2]. A construção de um circuito integrado pode ser feita em diferentes escalas de integração, podendo chegar a uma escala de integração na qual em um único chip sejam construídos mais de 10 milhões de componentes, denominada ULSI (Ultra Large Scale Integration) [2]. Os circuitos integrados comumente utilizados na construção de circuitos eletrônicos digitais apresentam uma escala de integração na qual é possível que até 30 dispositivos sejam construídos em um único chip, denominada SSI (Small Scale Integration) [2]. Esses circuitos integrados são encapsulados de diversas maneiras e o formato conhecido como DIP (dual-in-line package) é o formato padrão utilizado, onde os pinos de conexão são dispostos em duas fileiras paralelas ao longo das laterais do corpo do circuito integrado, que possui o formato de uma pastilha [2]. É através dos pinos de conexão que os circuitos internos do Circuito Integrado se comunicam com os circuitos externos. Ainda na pastilha existe um entalhe que tem a função de servir como referência na identificação da numeração dos pinos de conexão. Um circuito integrado pode ser referenciado pela sigla “CI”, seguida do número de identificação desse circuito integrado. No caso da Figura 2.7 o CI 7408 e o CI 7483 são apresentados.

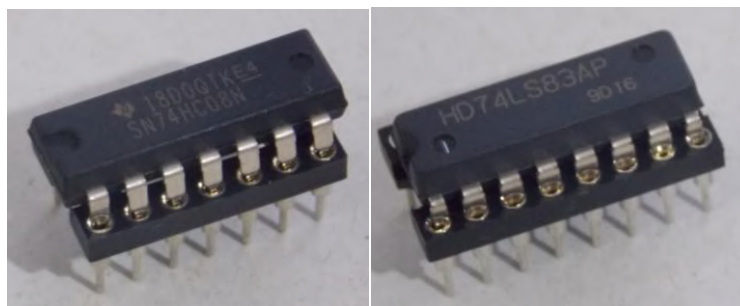


Figura 2.7 – CI 7408 e CI 7483.

Um Circuito Integrado utilizado na montagem de um circuito eletrônico digital pode conter em seu interior um circuito eletrônico que reproduz as operações da Álgebra de Boole,

como é o caso do CI 7408, que apresenta quatro operações de produto lógico independentes entre si. No interior do CI 7408 existem quatro circuitos eletrônicos, onde cada um deles realiza a operação de produto lógico com os sinais de entrada de dois pinos, impondo o sinal correspondente a resposta dessa função a um terceiro pino. Os circuitos internos de um Circuito Integrado são construídos através de técnicas industriais de encapsulamento onde todos os componentes do circuito são fabricados num mesmo processo sobre um material semicondutor de silício. Esses circuitos diferem dos circuitos eletrônicos discretos, onde os componentes eletrônicos isolados são interligados, uma vez que não é possível desmontar os circuito e reaproveitar os seus componentes, e mesmo que a pastilha do Circuito Integrado seja aberta e seus circuitos internos expostos, é difícil identificar quais são e onde estão os componentes do circuito.

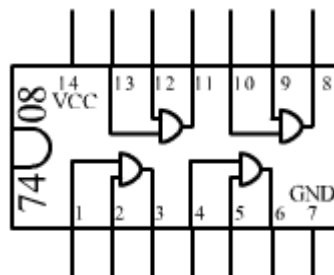


Figura 2.8 – Representação do CI 7408.

No caso do CI 7408, cada um de seus quatro circuitos pode ser representado pelo operador referente à operação lógica que esse circuito executa. Assim, é possível observar na Figura 2.8 que no interior da pastilha do CI 7408, foram inseridas quatro imagens que simbolizam a operação de produto lógico, e cada uma delas teve suas entradas a sua saída ligadas aos pinos do circuito integrado referentes ao circuito que representam.

Existem Circuitos Integrados que podem conter em seu interior um circuito eletrônico mais complexo, capaz de executar funções mais sofisticadas, como é o caso do CI 7483, que tem como função realizar a operação de soma algébrica entre dois números de quatro dígitos. Neste caso, são inseridos textos próximos aos pinos para indicar o que cada pino representa.

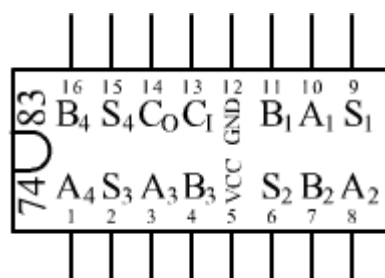


Figura 2.9 – Representação do CI 7483.

As representações apresentadas nas Figuras 2.8 e 2.9 são puramente didáticas e têm finalidade de facilitar o reconhecimento da função de cada pino do circuito integrado durante o estudo dos mesmos. Os circuitos integrados reais apresentam apenas o seu número de identificação impresso sobre sua pastilha.

Os circuitos integrados podem ser classificados quanto ao tipo de transistores utilizados em sua construção: Bipolar ou Mosfet [2], e nesses dois grupos eles são agrupados em famílias lógicas. Entre os Bipolares a família TTL (Transistor-Transistor Logic) pode ser citada. Entre os Mosfet a família CMOS (Complementary MOS) pode ser citada. Os circuitos integrados da família lógica TTL possuem número de identificação iniciados pelos números “74” e “54”, e os circuitos integrados mais relevantes da família lógica CMOS, pelo número “40”. Os Circuitos Integrados da família TTL foram os selecionados para fazer parte da biblioteca do programa

O programa desenvolvido possui uma biblioteca que contém os seguintes Circuitos Integrados: 500, 507, 560, 2114, 7400, 7402, 7404, 7408, 7410, 7411, 7413, 7417, 7420, 7421, 7427, 7428, 7430, 7432, 7437, 7440, 7442, 7448, 7474, 7476, 7483, 7485, 7486, 7490, 7493, 74125, 74151, 74153, 74175, 74181, 7194 e 9368. Toda a informação sobre o funcionamento destes Circuitos Integrados foi obtida através das folhas de dados fornecidas por diversos fabricantes [3], [4] e [5]. Além dos Circuitos Integrados em suas formas de pastilha, estão disponíveis na biblioteca do programa as seguintes portas lógicas: AND, NAND, OR, NOR, EXOR, EXNOR, NOT, BUFFER, TRI-STAGE, MUX, DEMUX, FLIP-FLOPS e MEMÓRIAS. Algumas dessas portas lógicas não representam um Circuito Integrado real, são apenas modelos didáticos, como é o caso do EXOR de três e quatro entradas. Existem também displays hexadecimais, displays de sete segmentos e LEDs. As fontes de sinal presentes são as chaves lógicas, portas lógicas de tipo Pulsers, fonte de tensão e pontos de aterramento. Essas portas lógicas são interpretadas pelo programa como sendo um circuito integrado em forma simplificada, onde existem apenas os pinos de entrada e saída referentes à porta lógica. Assim todas as futuras explicações aplicadas aos circuitos integrados também valem para essas portas lógicas.

Capítulo 3

Análise do Circuito

3.1. Circuito Digital

Em um circuito eletrônico digital, cada circuito integrado deve ter o pino nomeado de “VCC” ligado a uma fonte de tensão, e também deve ter o pino nomeado de “GND” ligado a um ponto de aterramento. Esse procedimento é necessário para que o Circuito Integrado possa ser energizado e seus componentes internos polarizados. Não se deve inverter essas ligações, pois há a possibilidade de danificar o Circuito Integrado. Para a família TTL, a fonte de tensão deve ser de 5V [5].

Os circuitos integrados se comunicam através de seus pinos, que são ligados por meio de fios condutores. Os circuitos internos do Circuito Integrado impõem aos pinos valores de tensão que representam estado lógico zero ou estado lógico um. Para a família TTL, o estado lógico zero é reconhecido para a faixa de tensão entre 0V e 0,8V em entrada de circuito e para a faixa de tensão entre 0V e 0,4V em saída de circuito [5]. O estado lógico um é reconhecido para a faixa de tensão entre 2V e 5V em entrada de circuito e para a faixa de tensão entre 2,4V e 5V em saída de circuito [5]. Caso o valor de tensão em um pino não esteja entre as faixas descritas, é dito que a tensão está em uma faixa proibida, assim o Circuito Integrado não funcionará corretamente, pois não consegue definir qual estado lógico esse pino apresenta.

Quando a tensão de um pino é alterada para os valores padrões de estado lógico um ou estado lógico zero, outro pino que estiver ligado a ele sentirá essa mudança de estado lógico e assim o circuito integrado ao qual faz parte poderá ter o estado lógico de suas saídas modificado. O circuito integrado que foi excitado pela mudança do estado lógico do pino, leva algumas frações de segundo para que, após o sinal de excitação se propagar pelos seus circuitos internos, haja a mudança do estado lógico de outro de seus pinos e essa mudança só ocorrerá se a excitação for suficiente para que a resposta da função do Circuito Integrado seja modificada.

A interação entre os circuitos integrados ocorre como descrita acima, com um excitando o outro após a mudança dos estados lógicos dos seus pinos, assim esse processo pode resultar em uma sucessão de excitações que se propagam pelos demais Circuitos Integrados do circuito. Essa sucessão de excitações tem início quando a excitação do primeiro Circuito Integrado resulta na mudança do estado de um de seus pinos, o que resulta na excitação de outro Circuito Integrado que tenha ligação com esse pino, assim esse último Circuito Integrado poder ter o estado lógico de um de seus pinos alterado, excitando um terceiro Circuito Integrado ligado a esse pino, e assim o processo de excitação é propagado pelos demais Circuitos Integrados que compõem o circuito. Dessa maneira é possível modificar o estado dos pinos de Circuitos Integrados em um circuito lógico apenas modificando o estado de um pino.

Para montagem de circuitos que apresentem características de realimentação, é possível chegar a um estado de instabilidade em que uma excitação em um Circuito Integrado pode provocar uma sucessão de excitações nos demais Circuitos Integrados, e devido a realimentação presente na montagem, o circuito acaba auto excitando-se, provocando a situação em que, após certos intervalos de tempo, um mesmo pino tenha seu estado lógico modificado sem interferência externa ao circuito.

Existem alguns itens a serem analisados na dinâmica da interação entre os circuitos integrados. O primeiro deles é o tempo de propagação. Como foi descrito no capítulo 2, no interior de um circuito integrado existe um circuito eletrônico composto principalmente por transistores, diodos, resistores e capacitores, e para que o sinal de excitação enviado por outro circuito integrado seja efetivamente traduzido em mudanças nos estados lógicos dos pinos de saída, ele deve primeiro se propagar por esse circuito e isso demora algumas frações de segundos. Na família TTL esse tempo de propagação é entorno de 9ns [5]. Esse tempo de propagação deve ser levado em consideração no projeto de circuitos, pois para o caso de um circuito composto por vários circuitos integrados ligados entre si, e que tenha uma resposta que dependa da propagação de um sinal por todos eles, essa resposta pode demorar certo tempo para ser atualizada, e até apresentar resultados intermediários antes de apresentar o resultado final.

O outro item a ser analisado é o fato de que existem limites para a quantidade de corrente que pode ser fornecida ou drenada por um pino de Circuito Integrado. Os circuitos internos de um Circuito Integrado possuem limites operacionais de condução de corrente, e se estes forem

ultrapassados podem resultar em dano ao Circuito Integrado. Uma vez que os pinos que representam entradas necessitam de valores específicos de corrente para que os estados lógicos zero e um sejam mantidos sobre eles, um pino que represente uma saída deve ser capaz de suprir a necessidade de corrente de todos os pinos ligados a ele, para que haja um funcionamento correto do circuito. Para realizar o cálculo do número de pinos de entrada ligados a um pino de saída, primeiramente define-se como Unidade de Carga os valores de 40 μA para estado lógico um e 1,6 mA para estado lógico zero, define-se como FAN-OUT o número de Unidades de Cargas que um pino que represente uma saída pode suportar, e definir-se como FAN-IN o número de Unidades de Carga que um pino que represente uma entrada necessita para manter seu estado lógico [5]. No caso do CI 7400, pinos que representam saídas de função, quando em estado lógico zero podem drenar um máximo de 16 mA e quando em estado lógico um podem fornecer um máximo de 800 μA , por sua vez, um pino que represente uma entrada, quando em estado lógico zero fornece 1,6mA e quando em estado lógico um drena 40 μA [5]. Com os valores listados é possível concluir que o FAN-IN do CI 7400 é de uma Unidade de Carga para estado lógico zero e um, e que seu FAN-OUT é de dez Unidades de Cargas para estado lógico zero e vinte Unidades de Cargas para estado lógico um. Assim é possível concluir que em uma saída de um CI 7400 podem ser conectadas a dez entradas, que possuam FAN-IN de no máximo uma Unidade de Carga para estado lógico zero, e FAN-IN de no máximo duas Unidades de Carga para estado lógico um. Os valores de FAN-IN e FAN-OUT calculados para o CI 7400 são os valores presentes em grande parte da família 74, porém existem Circuitos Integrados que possuem valores de FAN-IN e FAN-OUT diferentes, como é o caso do CI 7474, em que o pino que representa a entrada Clock fornece 3,2 mA quando em estado lógico zero, e drena 80 μA quando em estado lógico um, ou seja, possui FAN-IN de duas Unidades de Carga para estado lógico zero e um. Assim é importante realizar o cálculo desses valores em cada pino para evita qualquer tipo de dano ou mau funcionamento dos circuitos integrados.

3.2. Tradução Computacional

No capítulo 3.1 foram descritos os detalhes de como um circuito eletrônico digital funciona e os tópicos que devem ser levados em consideração em seu projeto. Porém para a tradução computacional do funcionamento desse circuito, algumas considerações foram necessárias.

O programa desenvolvido foi criado para ser um simulador lógico, no qual cada elemento do circuito possui uma função lógica que depende dos estados de suas entradas. Não existe qualquer tipo de análise de variáveis de circuitos eletrônicos, como tensão e correntes nos pinos. Assim as considerações feitas sobre as tensões de alimentação dos circuitos integrados e os níveis de tensão que representam cada estado lógico não são levadas em conta no desenvolvimento do programa. Essas tensões são substituídas por variáveis que apresentam valor zero para estado lógico zero ou valor um para estado lógico um.

Não existem limites de ligações em um mesmo pino, assim como foi descrito no capítulo 3.1. Uma vez que não há cálculo das correntes que entram e saem de cada pino dos Circuitos Integrados, os valores de FAN IN e FAN OUT não são levados em consideração. É permitido realizar quantas ligações sejam necessárias em um mesmo pino, a única restrição existente é que somente a primeira das ligações pode ser feita diretamente sobre o pino em questão e as demais devem ser realizadas sobre o fio utilizado na primeira ligação.

Uma vez que um circuito esteja montado e a simulação seja executada, ela é feita de maneira iterativa. A cada iteração, todos os Circuitos Integrados têm o estado lógico de suas saídas atualizado com base nos estados lógicos que suas entradas apresentavam no início da iteração. Assim o tempo de propagação de cada circuito integrado é desconsiderado, uma vez que o período da execução da simulação é fixado pelo programa e ao final de um período todos os circuitos integrados exibem os novos estados de suas saídas. O tempo que uma excitação levará para ser propagada a todos os elementos do circuito depende do período de simulação do programa.

Como foi descrito, os circuitos integrados apresentam em seus pinos estados lógicos que são tratados como variáveis pelo programa, e os estados de suas saídas, que também são tratadas como variáveis, são determinadas após o programa analisar as variáveis que representam os estados lógicos das entradas. Cada Circuito Integrado possui um conjunto de instruções para o programa realizar a análise de seus pinos de entrada e mostrar nas saídas os

estados lógicos correspondentes a essas entradas. No caso do CI 7408, as instruções são para que o programa analise as variáveis correspondentes aos estados lógicos dos dois pinos de entrada de cada operação de produto lógico e faça com que a variável correspondente ao estado lógico do pino de saída de cada uma dessas operações tenha valor um apenas se ambas as variáveis de entrada apresentem valor um. As instruções de análise dos pinos de entrada são baseadas nas funções lógicas as quais esses pinos fazem parte.

No caso dos Circuitos Integrados que contenham portas lógicas de tipo Flip-Flop, a definição das instruções de análise dos pinos necessitou a utilização de um fluxograma. Essa abordagem foi necessária para que as características de memória que esse tipo de circuito apresentam pudessem ser criadas na simulação. Assim para cada flip-flop em um circuito integrado foi criado um fluxograma de estados e as instruções de análise, além de levar em consideração os pinos de entrada também contêm informações sobre o fluxograma de estados do flip-flop.

Para os Circuitos Integrados que representam memórias, foi necessário criar um conjunto de variáveis que representassem as posições da memória e seu conteúdo. Essas variáveis são acessadas pelo programa e podem ser alteradas. As instruções de análise dos pinos desses circuitos integrados fazem com que o programa acesse essas variáveis e utilize seus valores para atualizar o estado dos pinos de saída, ou modificar os valores armazenados nessas variáveis.

Existem alguns casos especiais de circuitos integrados com funções complexas, em que a definição das instruções de análise dos pinos representaria um imenso trabalho de projeto, assim, a solução adotada foi de o programa criar e simular o circuito equivalente composto apenas de portas lógicas básicas. Assim as portas lógicas que compõem o circuito equivalente têm suas instruções de análise seguidas pelo programa, e após todos serem analisados, os pinos do circuito integrado que contêm essas portas lógicas têm suas variáveis de estado lógico alteradas. Essa solução foi implementada para o CI 7490, CI 7493, CI 74175 e CI 74181, porém foi necessário abortar a utilização destes uma vez que são necessárias muitas modificações para que o programa possa trabalhar sem apresentar erro.

O fenômeno do curto-circuito foi abordado no desenvolvimento do programa de maneira a permitir sua existência, porém com algumas considerações. O fenômeno do curto circuito ocorre quando há a passagem de uma corrente elétrica de altíssima amperagem por um circuito onde há uma pequena resistência. Geralmente esse fenômeno ocorre quando

acidentalmente dois pontos de um circuito com diferentes valores de tensão são conectados por um fio condutor que apresenta baixo valor de resistência. Para o caso do programa o curto-circuito é identificado quando dois ou mais pinos que representem saída de um Circuito Integrado são ligados por um fio. O conjunto de pinos que participa do curto circuito fica marcado pelo programa e assim durante as iterações da simulação uma análise é feita sobre esses pinos. A análise primeiramente prioriza a existência das fontes de sinal, as chaves lógicas, portas lógicas de tipo Pulsers, fonte de tensão e pontos de aterramento. Caso somente uma fonte de sinal participe do curto circuito, seu estado lógico prevalecerá sobre os estados lógicos dos demais pinos do nó em curto, porém se existir mais de uma fonte de sinal, o estado lógico um só será o estado final do nó em curto, se todas as fontes de sinal apresentarem estado lógico um, caso apenas uma apresente estado lógico zero, o estado lógico final do nó em curto será zero. No caso de não existirem fontes de sinal, a análise é feita sobre todos os pinos de saída que participam do curto, e somente se todos eles apresentarem estado lógico um, o estado final do nó em curto será um, caso contrário o estado final do nó em curto será zero. O estado final do nó em curto é aquele que excitará os pinos que representem entradas de função que estejam ligados a esse nó.

As considerações feitas sobre o funcionamento de circuitos que apresentam situações de curto-circuito foram feitas para que esse tipo de ligação pudesse ser realizado no programa, porém não é recomendado realizar esse tipo de ligação em um circuito real, pois é possível danificar os componentes do circuito, assim todas as vezes que uma ligação desse tipo for realizada no programa, uma mensagem de aviso é exibida informando que uma situação de curto-circuito foi criada.

Capítulo 4

Descrição do Programa

4.1. Interface Gráfica

A interface gráfica do programa foi projetada tendo como base a simplicidade de acesso às funções do mesmo, assim não são utilizados menus e sub-menus com texto para acessar as funções, e em seu lugar são utilizados somente botões que representam essas funções.

A janela do programa é dividida em quatro regiões. A primeira região é a Barra de Ferramentas, que fica localizada na parte superior da janela do programa. A segunda região é o Container de Portas que se localiza na barra a esquerda da janela do programa. A terceira região é a Barra de Informações que se localiza da parte inferior da janela do programa. A quarta região é a Área de Trabalho que se localiza entre as três regiões anteriores.

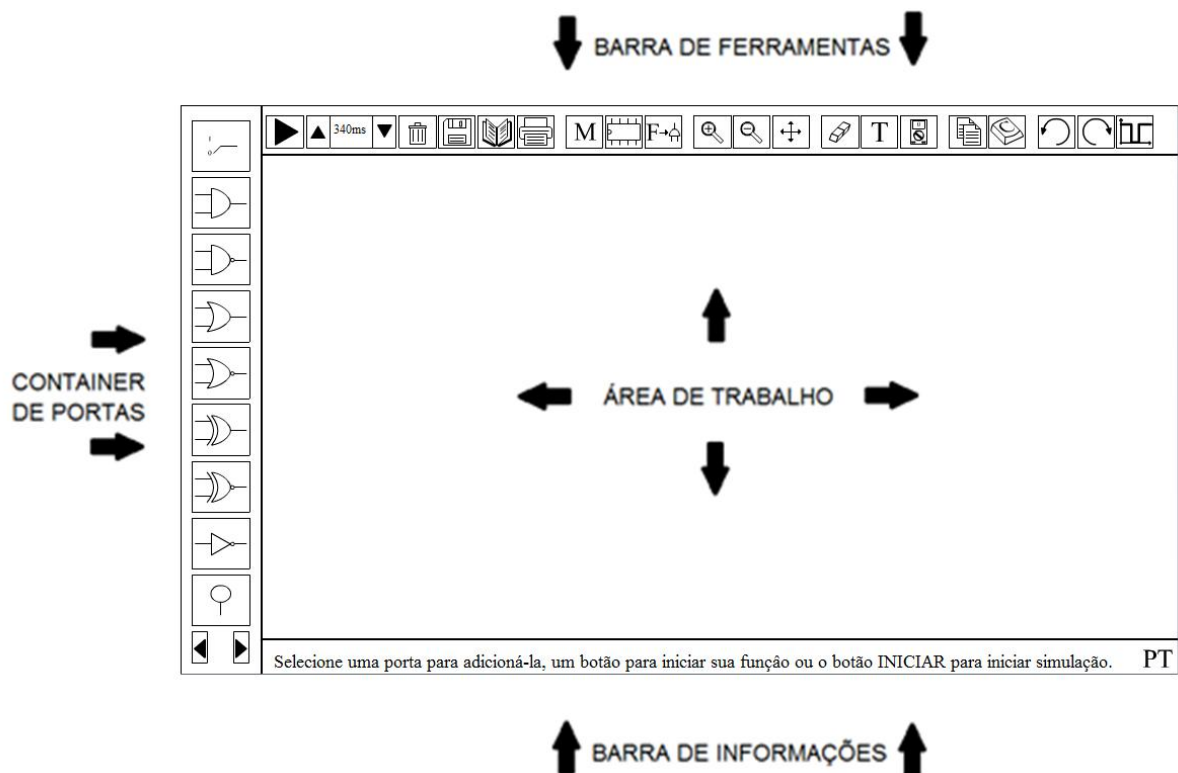


Figura 4.1 – Regiões da interface gráfica do programa.

4.1.1. Barra de Ferramentas

A Barra de Ferramentas contém todos os botões que representam as funções do programa e estes são pequenos quadrados com uma imagem em seu interior que ilustra a função desses ícones. A ativação das funções as quais os botões representam se dá após o usuário posicionar o cursor do mouse sobre um desses botões e pressionar o botão esquerdo no mouse. Alguns botões têm suas imagens alteradas quando sua função é ativada, porém retornam a imagem original assim que essa função é desativada. Com o fluxo do programa alguns botões podem ser bloqueados temporariamente, porém eles serão liberados assim que sua utilização for permitida dentro do programa.

Existem dezenove botões na barra de ferramentas e um grupo botões/mostrador. Abaixo será feita uma breve descrição das funções de todos os botões, na ordem em que estão dispostas na barra de ferramentas:

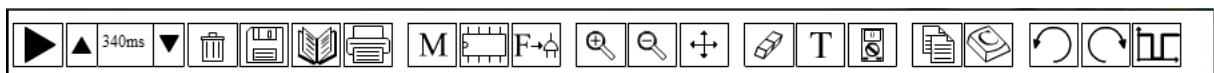


Figura 4.2 – Detalhamento da barra de ferramentas.

4.1.1.1. INICIAR/CANCELAR

Esse botão está associado a duas funções distintas, a primeira função é a de simulação do circuito e a segunda é a de cancelamento de uma função previamente selecionada. A função a ser iniciada por esse botão depende de qual imagem o botão exibe. Caso o botão apresente a imagem de uma seta direcionada para a direita, a primeira função será iniciada, porém se a imagem exibida for de um "X", a segunda função será iniciada.



Figura 4.3 – Imagens do botão INICIAR/CANCELAR.

Durante a simulação do circuito, o botão ainda poderá apresentar mais duas imagens que serão associadas a outras funções pertencentes à função de simulação.



Figura 4.4 – Imagens do botão INICIAR/CANCELAR durante simulação.

4.1.1.2. MAIS/MOSTRADOR/MENOS

Esse grupo de botões é composto por dois botões e um mostrador, e eles são associados a duas funções no programa. A primeira função é a edição do período de atualização da simulação do circuito, e essa função está disponível a todo o momento durante a execução do programa, a não ser quando o Editor de Minimização está ativo e sua tabela está adicionada na Área de Trabalho. Nesse caso a segunda função está disponível, que é a edição da quantidade de variáveis que serão analisadas no editor. O botão mais a esquerda, cuja imagem é a de uma seta direcionada para cima, ao ser selecionado aumenta o período ou a quantidade de variáveis e o botão mais a direita, cuja imagem é a de uma seta direcionada para baixo, ao ser selecionado diminui o período ou a quantidade de variáveis, enquanto o mostrador exibe o período em milissegundos ou a quantidade de variáveis do editor de minimização.

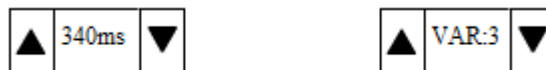


Figura 4.5 – Imagens do grupo MAIS/MOSTRADOR/MENOS.

4.1.1.3. DELETAR

Esse botão está associado à função de limpar todo o conteúdo do sistema criado pelo usuário. Essa função elimina da memória utilizada pelo programa todos os dados inseridos pelo usuário (portas lógicas e fios criados), porém mantém as informações de atalhos criados. Após o botão ser selecionado, sua imagem é alterada para indicar que a função foi ativada.



Figura 4.6 – Imagens do botão DELETAR.

4.1.1.4. SALVAR

Esse botão está associado à função de salvamento do conteúdo do sistema criado pelo usuário em um arquivo de dados. Ao ser selecionado chama a janela de salvamento e muda sua imagem para indicar que sua função se mantém ativada.

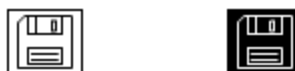


Figura 4.7 – Imagens do botão SALVAR.

4.1.1.5. ABRIR

Esse botão está associado à função de carregamento de um arquivo de dados que tenha sido criado previamente pelo programa. Quando esse botão é selecionado, chama a janela de carregamento de arquivo e muda sua imagem para indicar que a sua função se mantém ativa.



Figura 4.8 – Imagens do botão ABRIR.

4.1.1.6. IMPRIMIR

Esse botão está associado à função de impressão do conteúdo do sistema. A impressão pode ser tanto da Área de Trabalho do programa, quanto do interior do Editor de Macros. Quando esse botão é selecionado, inicia-se a função de impressão e a imagem do botão muda para indicar que essa função está ativa.



Figura 4.9 – Imagens do botão IMPRIMIR.

4.1.1.7. MINIMIZAR

Esse botão está associado ao Editor de Minimização. Quando selecionado, esse botão inicia o Editor de Minimização e muda sua imagem para indicar que o editor foi iniciado. Com o editor ativo, esse botão também executa outra função interna do editor.



Figura 4.10 – Imagens do botão MINIMIZAR.

4.1.1.8. MACROS

Esse botão está associado ao Editor de Macros. Ao ser selecionado, esse botão inicia o Editor de Macros e muda sua imagem para indicar que o editor está ativo. Com o editor ativo, esse botão também executa outra função interna do editor.



Figura 4.11 – Imagens do botão MACROS.

4.1.1.9. INTERPRETAR

Esse botão está associado ao Editor de Interpretação de Funções. Ao selecionar esse botão, o Editor de Interpretação de Funções é iniciado e a imagem do botão é substituída por uma que indica que o editor está ativo. Com o editor ativo, esse botão também executa outra função interna do editor.



Figura 4.12 – Imagens do botão INTERPRETAR.

4.1.1.10. AMPLIAR

Esse botão está associado à inicialização da função que permite ampliar/aproximar a Área de Trabalho, para ter uma visão mais detalhada de certa área do circuito. Ao ser selecionado, sua imagem muda para indicar que a função foi iniciada.



Figura 4.13 – Imagens do botão AMPLIAR.

4.1.1.11. REDUZIR

Esse botão está associado à inicialização da função que permite reduzir/afastar a Área de Trabalho, para ter uma visão global de todo o circuito. Ao ser selecionado, sua imagem muda para indicar que a função foi iniciada.



Figura 4.14 – Imagens do botão REDUZIR.

4.1.1.12. **ARRASTAR**

Esse botão está associado à inicialização da função que permite arrastar portas lógicas e fios na Área de Trabalho, ou arrastar a Área de Trabalho inteira. Ao ser selecionado, sua imagem muda para indicar que a função foi iniciada.



Figura 4.15 – Imagens do botão ARRASTAR.

4.1.1.13. **APAGAR**

Esse botão está associado à inicialização da função que permite retirar do sistema criado pelo usuário, portas lógicas e fios. Ao ser selecionado, sua imagem muda para indicar que a função foi iniciada.



Figura 4.16 – Imagens do botão APAGAR.

4.1.1.14. **TRACE**

Esse botão está associado à ativação da função de pintura dos fios baseado em seu estado lógico quando em simulação. Ao ser selecionado, sua imagem muda para indicar que a função foi iniciada.



Figura 4.17 – Imagens do botão TRACE.

4.1.1.15. **PROBE**

Esse botão está associado à utilização da ferramenta virtual PROBE, que tem a função de indicar o estado lógico de fios e de pinos das portas lógicas quando em simulação. Ao ser selecionado, sua imagem muda para indicar que a função foi iniciada.



Figura 4.18 – Imagens do botão PROBE.

4.1.1.16. DUPLICAR

Esse botão está associado à inicialização da função de duplicação, que cria uma cópia do grupo selecionado para ser adicionado ao sistema.



Figura 4.19 – Imagem do botão DUPLICAR.

4.1.1.17. ATALHOS

Esse ícone está associado ao Editor de Atalhos. Ao selecionar esse botão, o Editor de Atalhos é iniciado e a imagem do botão é substituída por uma que indica que o editor está ativo.



Figura 4.20 – Imagens do botão ATALHOS.

4.1.1.18. ROTAÇÃO ANTI-HORÁRIA

Esse botão está associado à inicialização da função de rotação do grupo selecionado em sentido anti-horário.



Figura 4.21 – Imagem do botão ROTAÇÃO ANTI-HORÁRIA.

4.1.1.19. ROTAÇÃO HORÁRIA

Esse botão está associado à inicialização da função de rotação do grupo selecionado em sentido horário.



Figura 4.22 – Imagem do botão ROTAÇÃO HORÁRIA.

4.1.1.20. GRÁFICOS

Esse botão está associado ao Editor de Gráficos. Ao selecionar esse ícone, o Editor de Gráficos é iniciado e a imagem do ícone é substituída por uma que indica que o editor está ativo.



Figura 4.23 – Imagens do botão GRÁFICOS.

4.1.1.21. DETALHES

Os botões AMPLIAR, REDUZIR e ARRASTAR (botões 4.1.1.10, 4.1.1.11 e 4.1.1.12 respectivamente), operam em conjunto como botões de rádio, ou seja, somente um pode estar ativo por vez, assim quando um dos botões estiver exibindo a imagem que indica que este se mantém selecionado e outro botão do grupo for selecionado, o primeiro volta a exibir sua imagem original e o último passa a exibir a imagem que indica sua seleção, desta maneira a função do último botão selecionado estará ativa.



Figura 4.24 – Grupo de botões operando como botões de rádio.

A seleção dos botões SALVAR e ABRIR (botões 4.1.1.4 e 4.1.1.5 respectivamente), como foram descrito em seus itens, inicializam as funções de salvamento e carregamento de arquivos de dados do sistema. A inicialização dessas funções faz com que o programa se comunique com o sistema operacional do computador para exibir as janelas de salvamento e carregamento de arquivos. Enquanto essas janelas não forem finalizadas pelo usuário, o programa se mantém travado a espera da resposta do sistema operacional.

Como foi descrito no início do capítulo 4.1.1, o processo de ativação das funções, as quais os botões representam, se dá após o usuário posicionar o cursor do mouse sobre um desses botões e pressionar o botão esquerdo no mouse, e esse processo é referenciado no texto deste trabalho como seleção do botão. Assim todas as vezes que o trecho “deve-se selecionar o botão” for encontrado no texto, deve-se entender que todo o processo descrito acima deve ser feito. Essa definição também se aplica às setas do Container de Portas.

4.2. Container de Portas

O Container de Portas é uma estrutura constituída de ícones representativos das portas lógicas disponíveis no programa e duas setas de mudança de página. Os ícones representativos das portas lógicas encontram-se organizados verticalmente dentro do Container de Portas, e são sempre dispostos em grupos de nove ícones, sendo onze o número de grupos de ícones presente na atual versão do programa. Abaixo do nono ícone encontram-se as setas, dispostas horizontalmente.

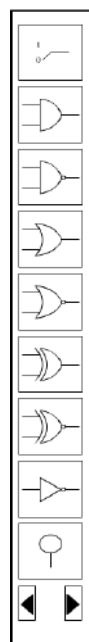


Figura 4.25 – Container de Portas.

Ao se selecionar uma das setas, elas realizam a mudança do grupo de ícones exibidos no Container de Portas, todos os nove ícones são substituídos por outros nove ícones. Cada grupo de ícones contém sempre os mesmos ícones. A seleção da seta que aponta para a direita fará com que o grupo exibido passe a ser o grupo que possui numeração igual à numeração do grupo atualmente exibido, acrescida de uma unidade. A seleção da seta que aponta para a esquerda fará com que o grupo exibido passe a ser o grupo que possui numeração igual à numeração do grupo atualmente exibido decrescida de uma unidade. Caso não haja grupo que possua numeração correspondente ao número do grupo calculado após a seleção das setas, o grupo que passa a ser exibido é o primeiro, caso o último grupo esteja sendo exibido, ou o último grupo, caso o primeiro grupo esteja sendo exibido.

A Figura 4.26 ilustra os onze grupos de ícones disponíveis no programa, na ordem em que são exibidos no programa.

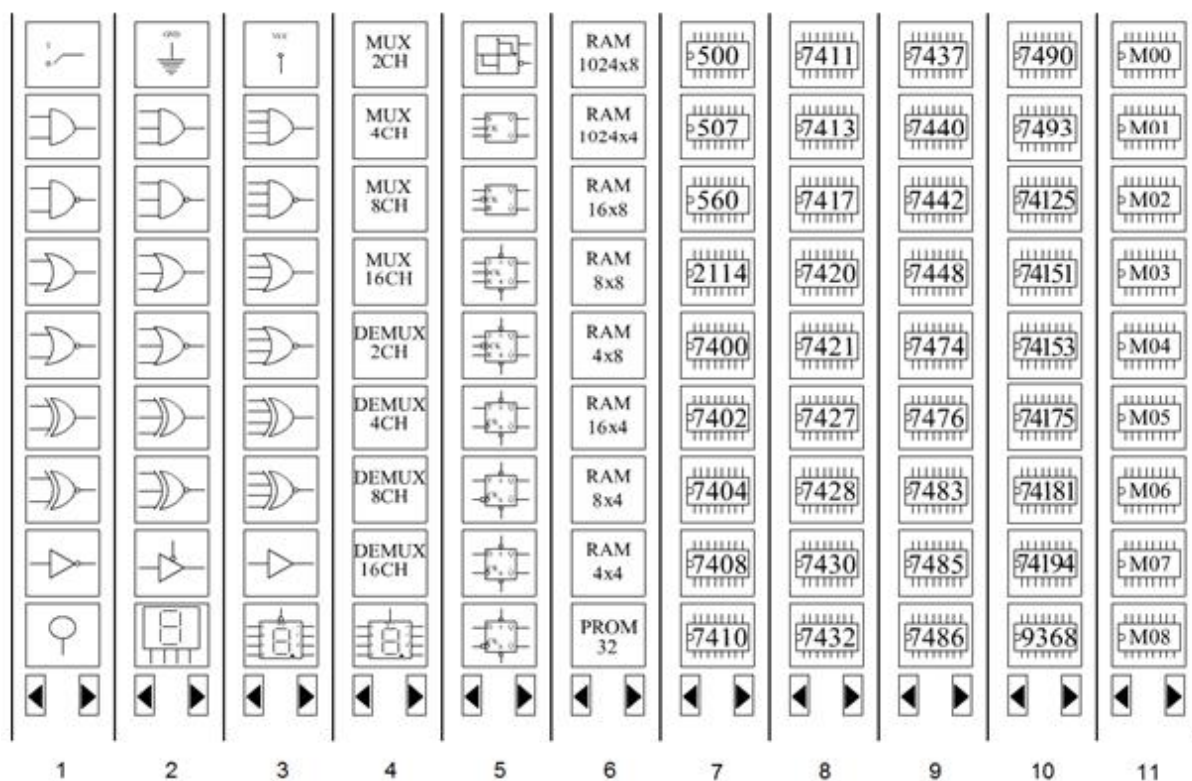


Figura 4.26 – Os onze grupos de ícones disponíveis no programa.

Cada um dos ícones dentro do Container de Portas representa uma porta lógica. A seleção de um desses ícones inicia a adição da porta lógica correspondente ao ícone selecionado na Área de Trabalho do programa.

Em certos pontos do programa, a seleção dos ícones do Container de Portas será bloqueada, pois a adição de portas lógicas ao sistema não será permitida. Assim que o programa retornar a um ponto onde seja possível adicionar portas lógicas ao sistema, a seleção desses ícones será liberada.

A regra de transição dos grupos de ícones no Container de Portas é absoluta durante a execução do programa, a não ser quando o Editor de Macros esteja ativo. Nesse caso a transição dos grupos de ícones, em qualquer direção, omitirá os grupos que possuem ícones representativos de circuitos integrados (grupos 7, 8, 9 e 10), uma vez que é proibida a inserção deste tipo de portas lógicas no sistema quando o Editor de Macros estiver ativo.

Ainda quando o Editor de Macros estiver ativo, o último grupo, que sempre contém os ícones referentes às portas lógicas de tipo Macros criadas pelo usuário no próprio Editor de

Macros, será substituído por um grupo de ícones especiais semelhantes aos ícones do grupo original. Esses novos ícones são diferentes dos anteriores, fazem parte do Editor de Macros e a seleção dos mesmos interage com o funcionamento deste editor. É possível observar na Figura 4.27 a mudança do último grupo de ícones, o grupo à esquerda é o grupo normal e o grupo à direita é o grupo que interage com o Editor de Macros, e nesse grupo um dos ícones apresenta imagem com fundo de cor preta.

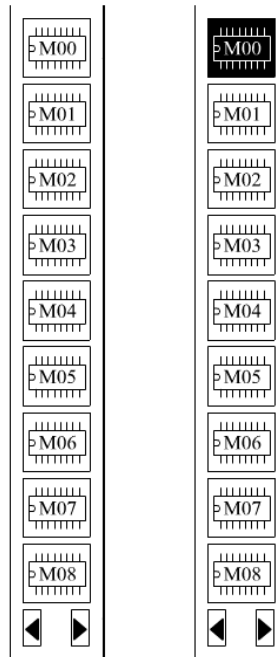


Figura 4.27 – Grupo de ícones de numeração onze.

4.3. Barra de Informações

A barra de informações tem a função de oferecer ao usuário do programa informações sobre as funções selecionadas, os editores ativos e sobre a função dos botões da barra de ferramentas.

Selecione uma porta para adicioná-la, um botão para iniciar sua função ou o botão INICIAR para iniciar simulação.

PT

Figura 4.28 – Barra de Informações.

As informações são sempre exibidas em forma de textos curtos que descrevem sucintamente as opções de ação do usuário, ou a função dos botões da barra de ferramenta. Caso o texto não possa ser exibido por completo, devido ao tamanho da janela do programa,

em alguns segundos o texto iniciará deslocamento em sentido à esquerda para que possa ser visualizado em sua totalidade.

No canto direito da barra de informações existe uma área que informa em qual idioma as informações serão exibidas. Existe um total de dois idiomas disponíveis: Português - brasileiro (PT) e Inglês (EN), e a seleção dessa área provoca a mudança do idioma entre os idiomas disponíveis.

4.4. Área de Trabalho

A Área de Trabalho é a região na qual as portas lógicas e fios são adicionados para se criar os circuitos a serem simulados. É também sobre a Área de Trabalho que as janelas dos editores do programa são adicionadas, sempre encobrindo os circuitos previamente criados.

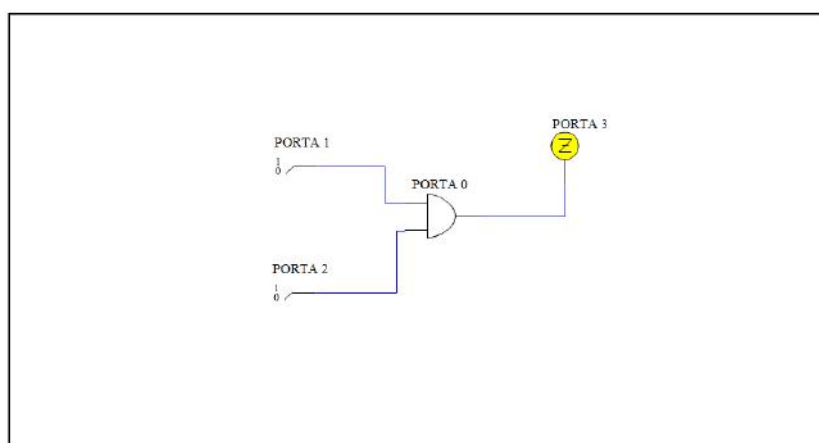


Figura 4.29 – Área de Trabalho.

A Área de Trabalho possui inicialmente tamanho de 6000 pixels de largura por 6000 pixels de altura. É possível alterar seu tamanho através de ferramentas do programa para que se possa visualizá-la com conforto em diferentes escalas, mas sempre mantendo a proporção de 1:1.

A janela do programa pode ser redimensionada livremente, porém as dimensões mínimas são de 1000 pixels de largura por 600 de altura e não há valores máximos para suas dimensões. Assim podem existir situações em que apenas parte da Área de Trabalho é exibida dentro da janela do programa. Nesses casos é possível arrastá-la, alterando sua posição dentro da janela do programa, para que se possa excursionar por toda sua área. Esse arrasto é sempre monitorado para não permitir que o usuário posicione a Área de Trabalho em um lugar onde não possa ver vista.

4.5. Funções

O programa possui quatorze funções, algumas voltadas a editar propriedades dos componentes dos circuitos na Área de Trabalho e outras voltadas à comunicação entre o programa e o sistema operacional.

As figuras que serão apresentadas nos próximos tópicos possuem um padrão. Elas são imagens retiradas da tela de um computador que executava o programa deste projeto, e apresentam trechos da interface gráfica do mesmo em diferentes momentos de sua execução. Cada figura pode apresentar apenas uma imagem, porém pode apresentar duas ou mais imagens. Quando duas imagens forem exibidas em uma mesma figura, o texto informativo da figura pode indicar o que cada imagem representa, porém existe outro caso em que essas imagens formam uma sequência de ações dentro do programa e a sequência de análise dessas imagens é sempre da esquerda para a direita, quando as imagens são dispostas lado a lado, ou de cima para baixo, quando as imagens são dispostas uma abaixo da outra. Nesses casos o texto informativo da figura não irá discriminar cada imagem da figura, irá informar a ação que essa sequência de imagens representa. Nos casos em que a figura possui mais de duas imagens, o texto informativo discrimina o que cada imagem representa.

4.5.1. Adição de Portas Lógicas

A adição de portas lógicas é o processo pelo qual é possível criar e inserir na Área de Trabalho novas portas lógicas, para que essas possam ser simuladas e seu funcionamento analisado. Esse processo também permite inserir portas lógicas no Editor de Macros para criar os circuitos desse editor. O processo de adição de Circuitos Integrados é idêntico ao processo de adição de portas lógicas, assim todas as explicações apresentadas neste capítulo para as portas lógicas valem para os Circuitos Integrados.

Para se realizar a operação de adição de portas lógicas deve-se selecionar no Container de Portas, através das setas do Container de Portas, o grupo de ícones que contém o ícone referente à porta lógica que se deseja inserir na Área de Trabalho. Em seguida deve-se posicionar o cursor do mouse sobre o ícone referente à porta lógica desejada, pressionar e soltar o botão esquerdo do mouse. Ao soltar o botão esquerdo, a porta lógica referente ao

ícone selecionado será criada e se movimentará junto ao cursor do mouse. O botão INICIAR/CANCELAR mudará sua imagem para a de um “X”, e a seleção desse ícone cancela o processo de adição da porta lógica.

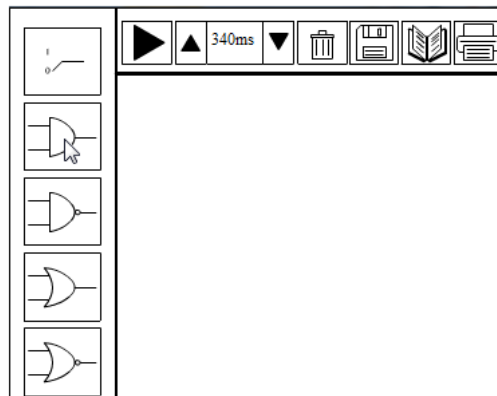


Figura 4.30 – Seleção do ícone desejado no Container de Portas.

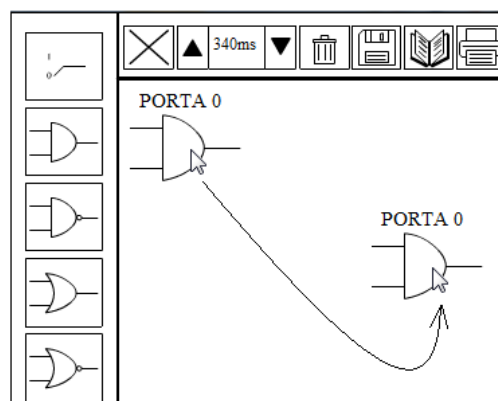


Figura 4.31 – Porta lógica criada acompanha o movimento do cursor do mouse.

Após a criação da porta lógica, deve-se posicionar o cursor do mouse, que agora traz consigo a mesma, sobre a Área de Trabalho e mais uma vez pressionar o botão esquerdo do mouse. Ao pressionar o botão a porta lógica será posicionada na Área de Trabalho na posição em que estava nesse momento e deixará de seguir o movimento do cursor do mouse. O botão INICIAR/CANCELAR volta a exibir sua imagem original. O processo de adição da porta lógica é encerrado.

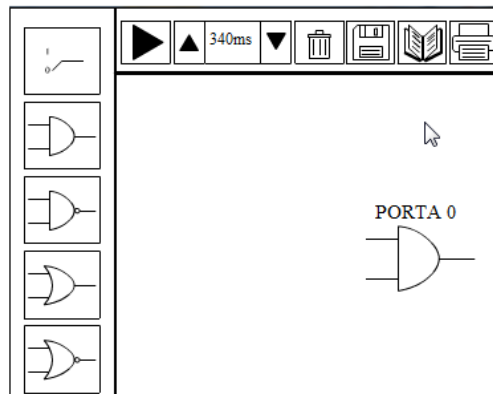


Figura 4.32 – A porta lógica adicionada deixa de seguir o movimento do cursor do mouse.

Existe uma regra a ser seguida no processo de adição de portas lógicas na Área de Trabalho. A regra é que não é permitido inserir portas lógicas de forma que alguma parte da mesma ultrapasse os limites da Área de Trabalho. Caso essa regra seja violada o processo de adição da porta lógica será cancelado e um aviso de erro será exibido na barra de informações, avisando o usuário que a posição da porta lógica não é permitida. Ocorrido o aviso o usuário deve informar novamente a posição desejada para a porta lógica na Área de Trabalho.

Como foi descrito no processo de adição de portas lógicas, após a seleção do ícone no Container de Portas, o botão INICIAR/CANCELAR passa a mostrar a imagem de um “X”, e caso este botão seja selecionado a adição da porta lógica será cancelada. Após a seleção deste ícone a porta lógica criada que acompanhava o movimento do mouse é retirada da Área de Trabalho e apagada do sistema do programa. Assim o botão INICIAR/CANCELAR volta a exibir sua imagem original e o processo de adição da porta lógica é finalizado.

Como a adição de portas lógicas está condicionada a seleção dos ícones do Container de Portas, nos momentos em que essa seleção estiver travada, a adição também estará travada. Os momentos em que a adição de portas lógicas encontra-se liberada são quando o programa não tem nenhuma outra função ou editor ativos, ou quando o Editor de Macros está ativo.

4.5.2. Criação de Fios

O processo de criação de fios possibilita a comunicação entre as portas lógicas do sistema, permitindo que, quando em simulação, os estados das saídas das portas lógicas possam ser sentidos pelas entradas de outras portas lógicas. A criação de fios no sistema depende da

existência de portas lógicas com pinos terminais livres, ou da existência de outros fios no sistema, e também está condicionada a situação em que nenhuma função ou editor estejam ativos, ou que apenas o Editor de Macros esteja ativo.

Para se iniciar o processo de criação de um fio, deve-se posicionar o cursor do mouse sobre um pino terminal livre de uma porta lógica, o que fará com que a Seta de Ligação desse pino seja exibida, ou sobre um fio existente, o que fará com que a cor do fio seja modificada para verde claro. Depois, pressionar e soltar o botão esquerdo do mouse. A Seta de Ligação é um elemento do programa criado para facilitar a seleção de um pino terminal de uma porta lógica. Ela possui o formato de um retângulo vermelho que fica junto do pino terminal, todos os pinos terminais possuem uma Seta de Ligação associada, porém elas não são exibidas até que o cursor do mouse seja posicionado próximo ou sobre o pino. A Figura 4.33 ilustra o início do processo de criação de um fio sobre um pino e sobre outro fio.

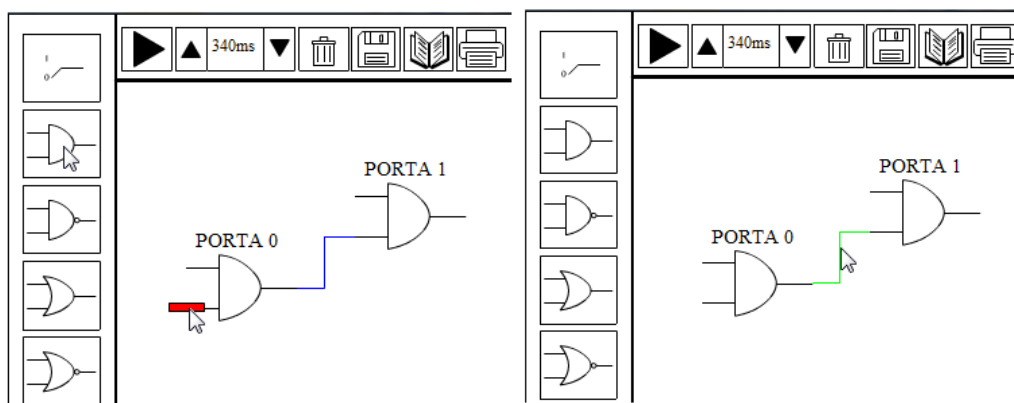


Figura 4.33 – Inicialização do processo de criação de fio, a esquerda sobre um pino e a direita sobre um fio.

Após soltar o botão do mouse, o fio é criado na Área de Trabalho e é iniciada a função de traçar fios. No caso do fio iniciado no pino de uma porta lógica, a Seta de Ligação deixa de ser exibida, e caso o fio tenha sido iniciado sobre outro fio, um Ponto Indicador de Ligação é adicionado sobre o ponto de ligação entre os dois fios para indicar a conexão entre esses dois fios. Ao movimentar o cursor do mouse, o fio criado será redimensionado para acompanhar esse movimento. O redimensionamento do fio se dará apenas no eixo vertical ou no eixo horizontal, e o eixo escolhido será feito pelo programa ao analisar o primeiro movimento do cursor do mouse depois de soltar o botão esquerdo do mesmo. Escolhido o eixo, todo movimento fora do mesmo será ignorado e apenas o movimento no eixo escolhido será traduzido em redimensionamento do fio. As Figuras 4.34 e 4.35 ilustram esse processo de redimensionamento do fio criado.

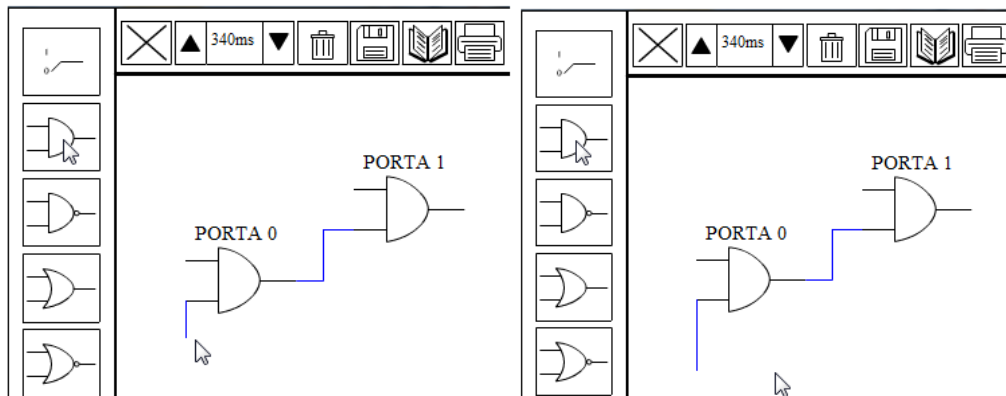


Figura 4.34 – Movimento do cursor do mouse é traduzido em redimensionamento do fio no eixo vertical.

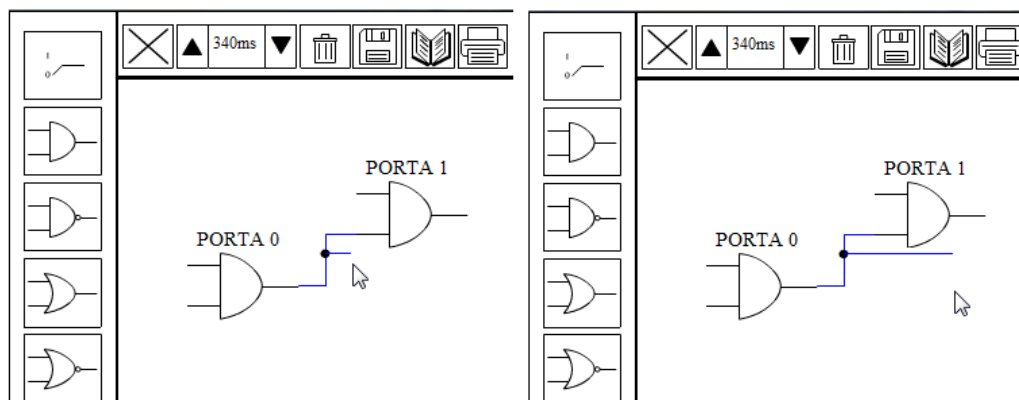


Figura 4.35 – Movimento do cursor do mouse é traduzido em redimensionamento do fio no eixo horizontal.

Iniciada a função de traçar fios, cada vez que o botão esquerdo do mouse for pressionado, estando o cursor do mouse sobre a Área de Trabalho, será inserida uma quina no fio e uma nova aresta será criada. A nova quina será posicionada de maneira que a aresta que era redimensionada mantenha o tamanho que possuía no momento em que o botão do mouse foi pressionado. A nova aresta terá orientação inversa em relação à aresta adjacente, ou seja, se a aresta adjacente tiver orientação vertical, a nova aresta terá orientação horizontal, e se a aresta adjacente tiver orientação horizontal, a nova aresta terá orientação vertical. A nova aresta passa a ser redimensionada para acompanhar o movimento do cursor do mouse.

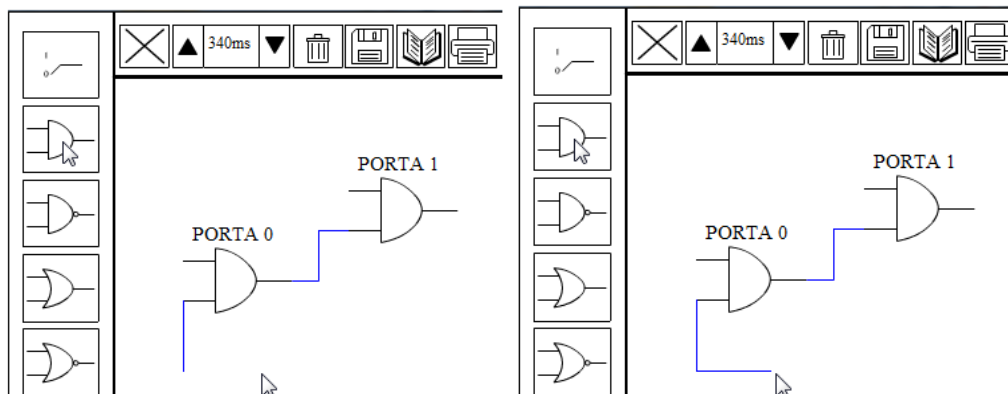


Figura 4.36 – Criação de nova aresta horizontal no fio.

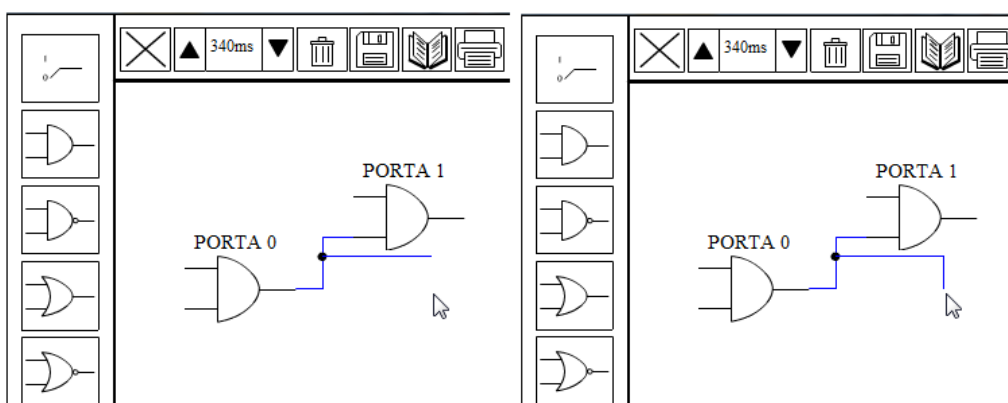


Figura 4.37 – Criação de nova aresta vertical no fio.

Podem ser inseridas quantas quinas sejam necessárias para formar o fio, repetindo o processo ilustrado na Figura 4.36 e Figura 4.37. Para finalizar o fio deve-se repetir o processo ilustrado na Figura 4.33, o que encerrará a função de traçar fios e completará a criação do fio.

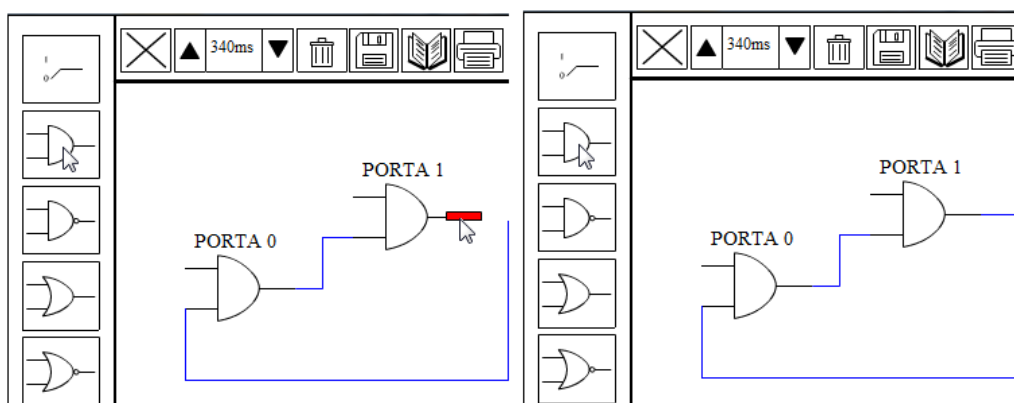


Figura 4.38 – Finalização do fio criado sobre um pino, selecionando a Seta de Ligação de uma porta lógica.

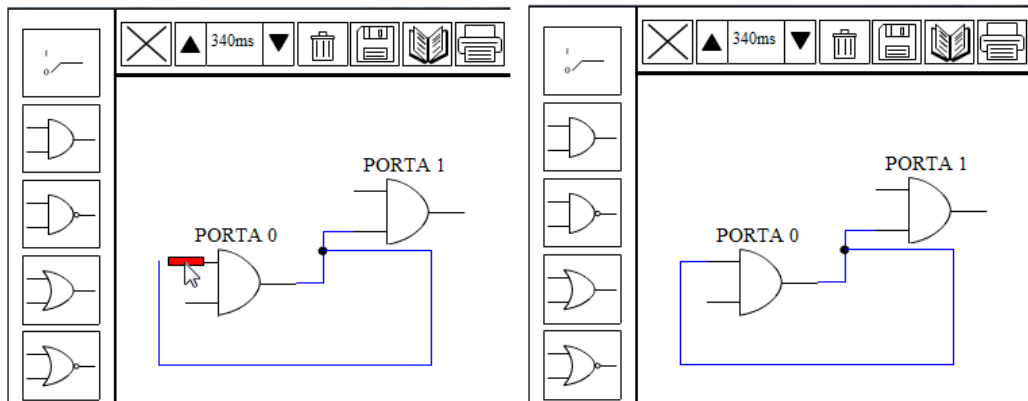


Figura 4.39 – Finalização do fio criado sobre outro fio, selecionando a Seta de Ligação de uma porta lógica.

A escolha da orientação da primeira aresta do fio é baseada no primeiro movimento realizado pelo cursor do mouse. A análise desse primeiro movimento leva em consideração em que região o primeiro movimento é realizado, se o mouse estiver nas regiões marcadas como “H” na Figura 4.40, a orientação será horizontal, e se estiver nas regiões marcadas como “V”, a orientação será vertical.

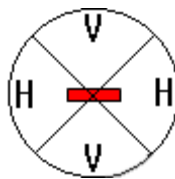


Figura 4.40 – Regiões de definição de orientação do fio criado.

O fio traçado pode iniciar ou terminar em qualquer pino terminal livre de qualquer porta lógica do sistema, assim como iniciar ou terminar sobre qualquer fio previamente traçado. Quando o fio inicia ou termina sobre um pino livre, esse pino muda seu status para ocupado e sua Seta de Ligação não é mais exibida quando se posiciona o cursor do mouse sobre esse pino. Essa mudança de status força o usuário a manter apenas um fio por pino. Quando o fio inicia ou termina sobre outro fio, um Ponto Indicador de Ligação é adicionado para criar uma referência visual de que há ligação entre estes dois fios. Esse tipo de ligação não possui limite como no caso dos pinos das portas lógicas, assim é possível ter tantos fios ligados a outro fio quantos sejam necessários.

Não é permitido que um fio tenha ligação com ele próprio, assim durante a função de traçar fios, não é possível selecioná-lo para encerrar o fio. É permitido traçar fios que resultem em qualquer tipo de realimentação entre entradas e saídas de uma mesma porta lógica. Também é permitido traçar fios que resultem em situações de curto-circuito.

Devido a problemas no projeto do programa, não é permitido criar fios com menos de quatro quinas, assim caso haja alguma situação em que ao tentar encerrar o fio, este possua menos de quatro quinas, uma informação de erro será exibida, e o usuário deverá inserir quinas até que o fio tenha pelo menos quatro.

Assim como na função de adição de portas lógicas, não é permitido traçar fios que ultrapassem os limites da Área de Trabalho, assim quando o cursor do mouse estiver fora dessa área, o fio parará de ser traçado e só voltará a acompanhar o cursor do mouse quando este voltar à Área de Trabalho.

Todo fio criado tem sua cor mudada para verde claro quando o cursor do mouse é posicionado sobre ele, assim como é ilustrado na Figura 4.33. Essa característica permite que um fio seja identificado em meio a um circuito com muitos fios, e encontra-se ativa enquanto nenhuma função ou editor estiver ativo, ou quando o Editor de Macros estiver ativo, ou quando as funções de arrasto e eliminação estiverem ativas. Os fios criados na janela do Editor de Interpretação de Funções também têm essa característica ativa. Assim que o cursor do mouse deixa de estar posicionado sobre o fio, ele retorna a apresentar a cor que possuía antes.

A qualquer momento depois de ter iniciado a função de traçar fios, pode-se cancelar a criação do fio. Para realizar esse cancelamento deve-se selecionar o botão INICIAR/CANCELAR que após o segundo item estará mostrando a imagem do "X". Assim, após cancelar a criação de fios, o fio criado é descartado, juntamente com o Ponto Indicador de Ligação que tenha sido inserido durante a criação do fio sobre outro fio e caso o fio tenha sido criado sobre um pino de uma porta lógica, sua Seta de Ligação deixa de apresentar status ocupado.

4.5.3. Simulação

A simulação do circuito na Área de Trabalho é iniciada após selecionar o botão INICIAR/CANCELAR quando o programa não estiver com nenhuma função ou editor ativo. Nessa situação o botão INICIAR/CANCELAR terá a imagem de uma seta direcionada para a direita. Após selecionar o botão, terá início a execução da simulação e a imagem exibida no botão INICIAR/CANCELAR muda para indicar que a simulação está ativa.

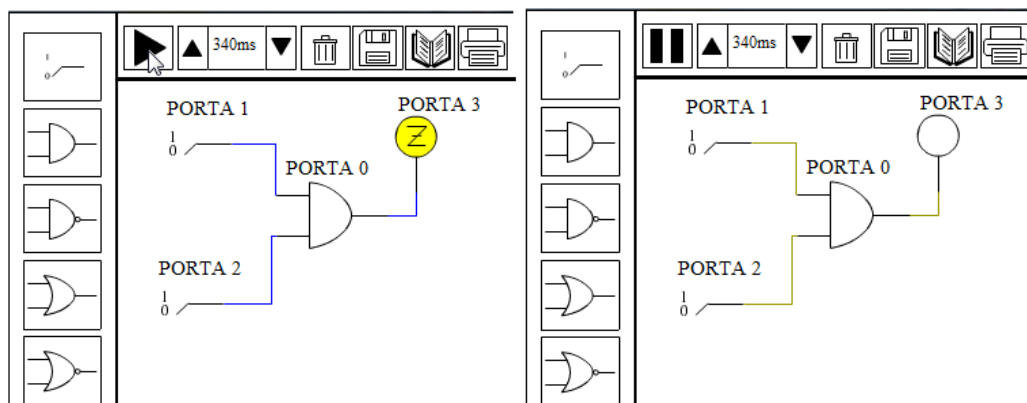


Figura 4.41 – Início da execução da simulação.

O período de atualização da execução da simulação é exibido no mostrador da barra de ferramentas, que estará exibindo o valor em milissegundos. Os botões com imagens de setas adjacentes ao mostrador podem editar seu valor entre um máximo de 3400 ms e um mínimo de 34 ms. A edição do valor da taxa de atualização não pode ser feita durante a simulação. O valor inicial do período de atualização da simulação é de 340 ms.

Durante a execução da simulação não é permitido adicionar portas lógicas à Área de Trabalho, assim a seleção dos ícones do Container de Portas se mantém bloqueada. A criação de fios também não é permitida durante a execução da simulação, assim as Setas de Ligação dos pinos das portas lógicas não são exibidas quando o cursor do mouse é posicionado sobre elas.

Uma vez que a simulação é iniciada somente as seguintes funções podem ser ativadas: função de ampliar e reduzir a Área de Trabalho; função de arrasto, restrita ao arrasto da Área de Trabalho; função Trace; função Probe e ativar o Editor de Gráficos. O Editor de Gráficos tem parte de suas funções bloqueadas durante a execução da simulação.

A função Trace é ativada automaticamente ao se iniciar a simulação, que faz com que a cor dos fios seja modificada dependendo do seu estado lógico, assim na Figura 4.41 a cor dos fios muda para indicar estado lógico zero.

Ao se iniciar a execução da simulação, todos os pinos das portas lógicas do circuito apresentam estado lógico de Alta Impedância. Esse estado representa a situação em que não há estado lógico sendo aplicado sobre esses pinos. Cada porta lógica interpreta esse estado de uma maneira diferente e depois da primeira iteração da simulação os pinos das portas lógicas passam a apresentar seus estados lógicos zero ou um, ou no caso das portas lógicas do tipo Tri-Stage e Memória, podem manter em seus pinos com estado de Alta Impedância. Os pinos

que representem entrada e não estejam ligados a nenhum outro pino de saída mantêm o estado lógico de Alta Impedância. Os circuitos na Área de Trabalho serão simulados e os estados lógicos dos pinos das portas lógicas serão calculados assim como foi descrito no capítulo 3.2.

As chaves lógicas são as portas lógicas que permitem ao usuário interagir diretamente com o circuito criado. Através delas é possível modificar o estado lógico das entradas das portas lógicas ligadas a essas chaves, através da modificação do estado lógico de suas saídas. Para se modificar o estado lógico da saída de uma chave lógica deve-se posicionar o cursor do mouse sobre esta chave lógica e pressionar o botão esquerdo do mouse, assim o estado lógico da chave será complementado e a imagem que representa essa porta lógica tem sua imagem modificada, passando a exibir uma que represente o novo estado lógico de sua saída. À medida que as iterações da simulação forem executadas, a modificação do estado lógico da saída da chave lógica fará com que as portas lógicas que tenham ligação com essa chave tenham os estados lógicos de suas saídas recalculados com os novos estados lógicos de suas entradas. As chaves lógicas são a única maneira de o usuário modificar diretamente o estado das entradas do circuito.

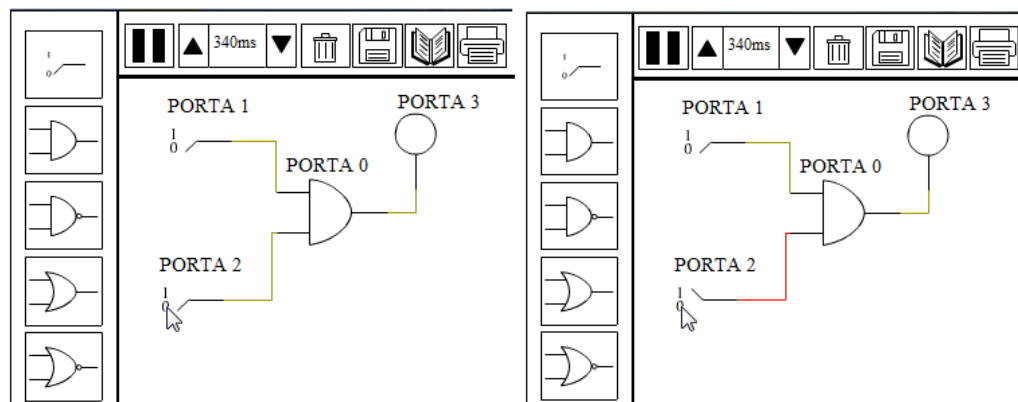


Figura 4.42 – Seleção da chave lógica provoca a mudança de seu estado lógico de saída.

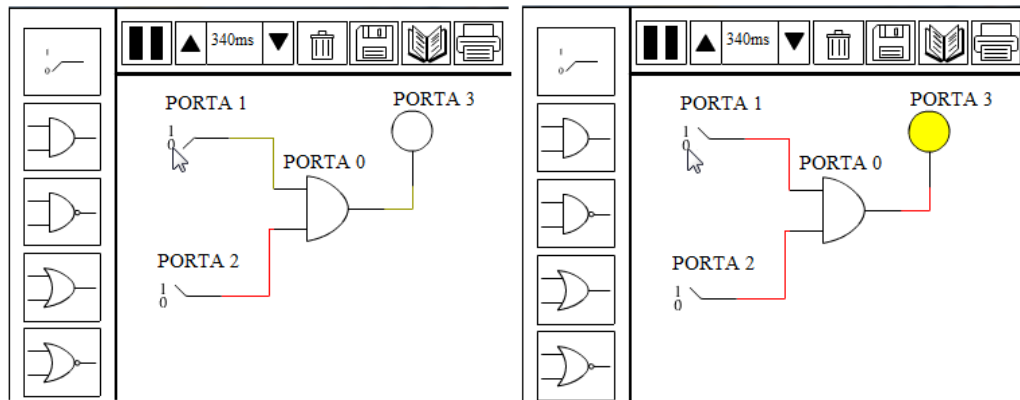


Figura 4.43 – Seleção da chave lógica muda o estado lógico de sua saída e o estado lógico da saída da porta lógica “PORTA 0”.

As Figuras 4.42 e 4.43 ilustram a modificação do estado lógico das chaves lógicas pelo usuário, primeiro a chave lógica “PORTA 1” e depois a chave lógica “PORTA 2”, e também mostram como essa modificação é interpretada pelo restante do circuito, com a modificação do estado lógico da saída da porta lógica “PORTA 0”.

Após o início da simulação, a imagem do botão INICIAR/CANCELAR exibida é modificada como mostrado na Figura 4.41. Ao selecionar o botão, a simulação é interrompida, e é possível ativar somente as mesmas funções que eram permitidas enquanto a simulação estava em execução, porém o Editor de Gráficos tem um número maior de funções disponíveis. Ao interromper a simulação, a imagem do botão INICIAR/CANCELAR muda novamente como mostrado na Figura 4.44. O botão INICIAR/CANCELAR possui duas funções distintas quando exibe essa imagem, a primeira é a de continuar a simulação e a segunda é a de encerrar a simulação. A imagem do botão é dividida em duas, e ao selecioná-lo com o cursor do mouse sobre a figura da seta apontando para a direita, a simulação é retomada, ou com o cursor do mouse sobre a figura do quadrado vermelho, a simulação é encerrada.

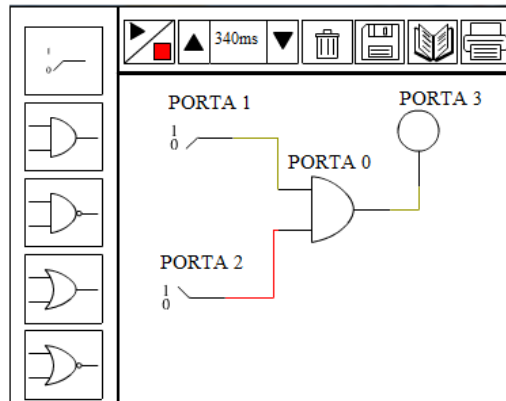


Figura 4.44 – Simulação interrompida.

Ao encerrar a simulação, todo circuito volta ao estado inicial onde o estado lógico de todas as entradas e saídas das portas lógicas e todos os fios é de Alta Impedância. O estado das chaves lógicas não é alterado, mantendo o estado que possuíam no momento em que a simulação foi encerrada. O botão INICIAR/CANCELAR volta a exibir sua imagem original.

4.5.4. Trace

A função Trace é responsável por gerar informação visual do estado lógico das entradas e saídas das portas lógicas durante a execução da simulação, através da mudança de cor dos fios ligados a essas entradas e saídas. As cores que os fios podem assumir estão associadas aos estados lógicos que as saídas e entradas das portas lógicas podem assumir.

A função Trace só pode ser ativada enquanto a simulação estiver em execução e ela é ativada automaticamente quando a simulação é iniciada. O modo de se ativar a função Trace é selecionar o botão TRACE. Assim a função é ativada e a imagem do botão muda para a imagem que indica que a função se mantém ativa. Para desativar a função deve-se selecionar o botão TRACE, agora exibindo a nova imagem, assim a imagem original volta a ser exibida e os fios passam a exibir cor azul. Caso a simulação seja encerrada enquanto a função Trace está ativa, essa função será desativada automaticamente.

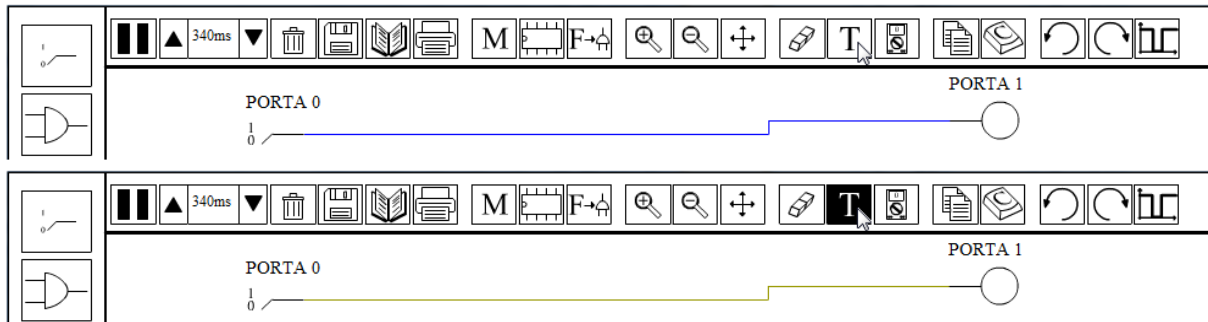


Figura 4.45 – Ativação da função Trace.

À medida que a simulação é executada, os pinos das portas lógicas têm seus estados lógicos atualizados, assim, havendo fios ligados a esses pinos, esses fios têm sua cor mudada para indicar o estado lógico do pino. Fio de cor azul representa Alta Impedância, fio de cor verde escuro representa estado lógico zero e fio de cor vermelha representa estado lógico um.

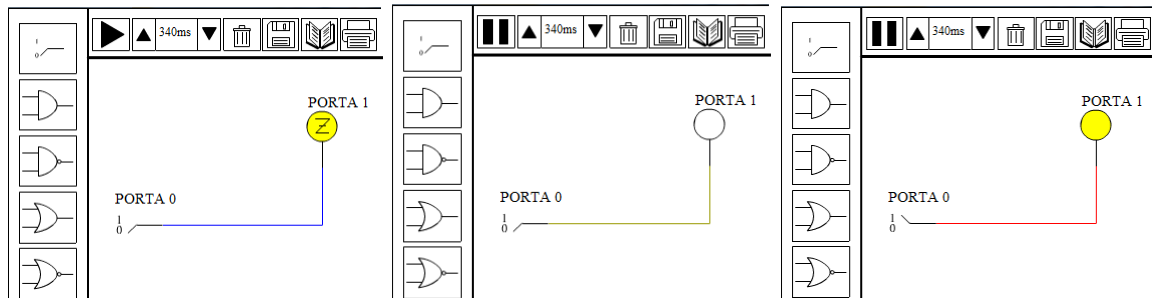


Figura 4.46 – Diferentes cores que um fio pode assumir.

A presença de fios de cor azul ligados a uma saída durante a execução da simulação com a função Trace ativa, indica que a porta lógica dessa saída não está impondo nenhum estado lógico nesse fio, ou seja, essa porta lógica pode ser considerada como inativa dentro do circuito. Uma segunda possibilidade seria o caso da primeira iteração da simulação, onde a informação do estado lógico do pino da porta lógica onde esse fio está ligado ainda não foi aplicada ao fio, assim ele mantém a cor azul, pois nesse fio ainda está presente o estado lógico de Alta Impedância. Após a primeira iteração da simulação, o fio mudará de cor para corresponder com o estado lógico do pino da porta lógica à qual está ligado.

Fios que não estejam ligados direta ou indiretamente a uma saída mantêm sempre a cor azul, uma vez que não há nenhuma porta lógica impondo estado lógico nesse fio. Caso a função Trace não esteja ativa, todos os fios do circuito se mantêm com a cor azul, mesmo que a simulação esteja sendo executada.

4.5.5. Probe

A ativação da função Probe é feita somente durante a execução da simulação, ao selecionar o botão PROBE, o que faz com que a imagem exibida no botão mude para uma que indica que a função se mantém ativa, e a imagem que representa a ferramenta virtual passa a acompanhar o cursor do mouse.

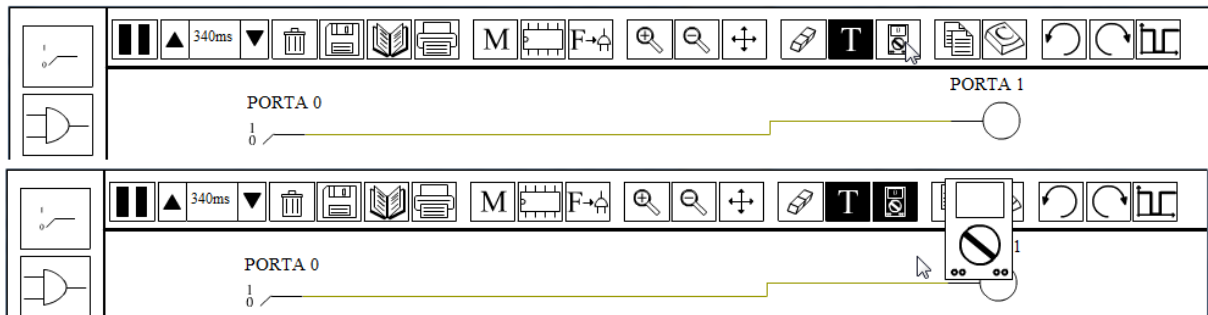


Figura 4.47 – Ativação da função Probe.

Esta ferramenta é capaz de medir o estado lógico dos pinos das portas lógicas e dos fios, exibindo no mostrador presente nessa ferramenta o valor zero para estado lógico zero, valor um para estado lógico um e valor “Z” para estado lógico de Alta Impedância. Enquanto a ferramenta não estiver medindo nenhum estado, seu mostrador permanece vazio. A Figura 4.48 ilustra os valores que podem ser exibidos na ferramenta virtual Probe.

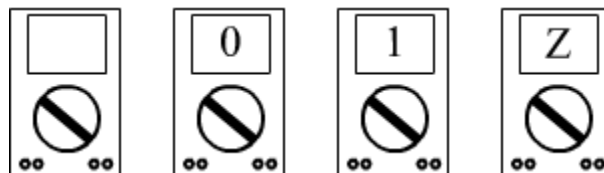


Figura 4.48 – As quatro possibilidades de informação a serem exibidas no mostrador da ferramenta virtual Probe.

Para realizar as medições deve-se posicionar o cursor do mouse sobre um fio ou sobre um pino de uma porta lógica, assim o mostrador da ferramenta exibirá o valor referente ao estado lógico medido.

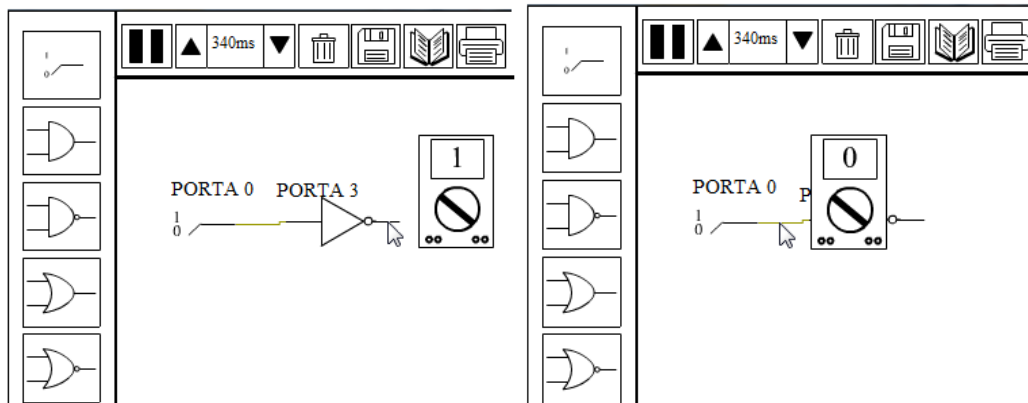


Figura 4.49 – Medição do estado lógico de um pino de uma porta lógica e de um fio.

A qualquer momento durante a execução da simulação, a função Probe pode ser desativada, selecionando o botão PROBE que exibe a nova imagem, o que resulta na retirada da ferramenta virtual Probe da Área de Trabalho e na exibição da imagem original do botão PROBE.

4.5.6. Seleção

A seleção de portas lógicas e fios na Área de Trabalho é uma função que é executada sem a necessidade de recorrer a nenhum botão. Para executar a seleção deve-se posicionar o cursor do mouse sobre uma porta lógica ou um fio e pressionar o botão direito do mouse. Após pressionar o botão, a porta lógica ou fio será selecionado e mudará de cor para indicar visualmente que faz parte do Grupo Selecionado. As portas lógicas têm sua cor mudada para amarelo e os fios para vermelho.

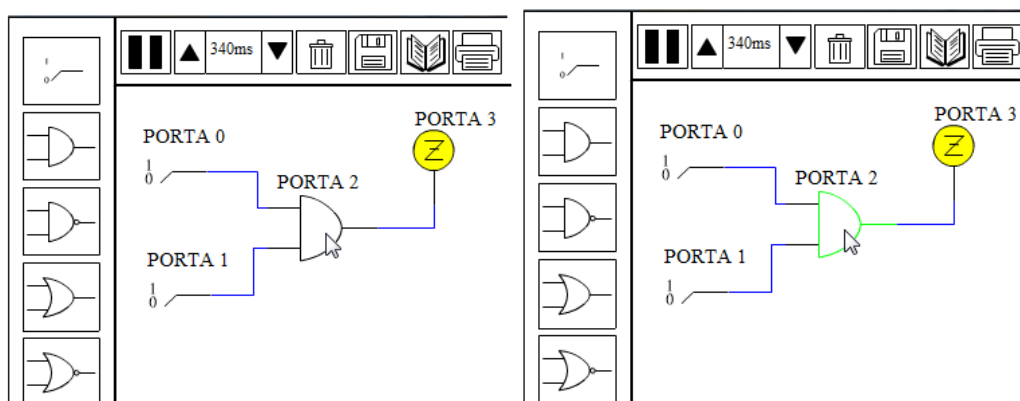


Figura 4.50 – Seleção de uma porta lógica.

Um fio, mesmo após ter sido selecionado e ter sua cor mudada para vermelho, ainda respeita o processo de identificação quando o cursor do mouse é posicionado sobre ele, assim como foi citado no final do capítulo 4.5.2. Sua cor é mudada para verde claro quando o cursor do mouse estiver posicionado sobre o fio e volta a ser vermelha assim que o cursor do mouse deixa de estar posicionado sobre este fio.

Caso outra porta lógica ou fio seja selecionado seguindo o processo de seleção descrito, enquanto houver alguma porta lógica ou fio selecionado, todos os elementos do Grupo Selecionado são excluídos do grupo e retornam a sua cor original, e somente a porta lógica ou fio selecionado permanece com a cor alterada e pertencente ao Grupo Selecionado.

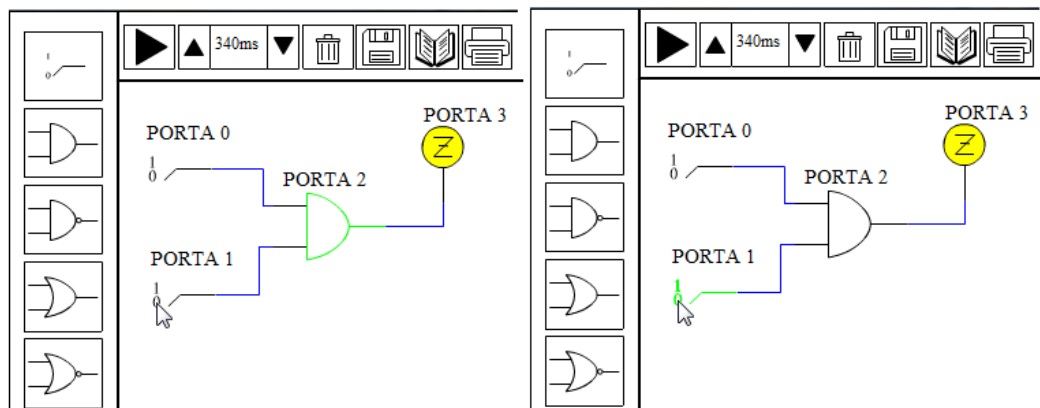


Figura 4.51 – Seleção de nova porta lógica para o Grupo Selecionado.

Para selecionar outras portas lógicas ou fios, e adicioná-las ao Grupo Selecionado, é necessário repetir o processo de seleção descrito, porém mantendo a tecla “SHIFT” pressionada durante todo o processo. Nesse caso as novas portas lógicas e fios selecionados mudam de cor e passam à do Grupo Selecionado, juntamente com as demais portas lógicas e fios que já faziam parte do Grupo Selecionado. Esse processo pode ser repetido para qualquer número de portas lógicas.

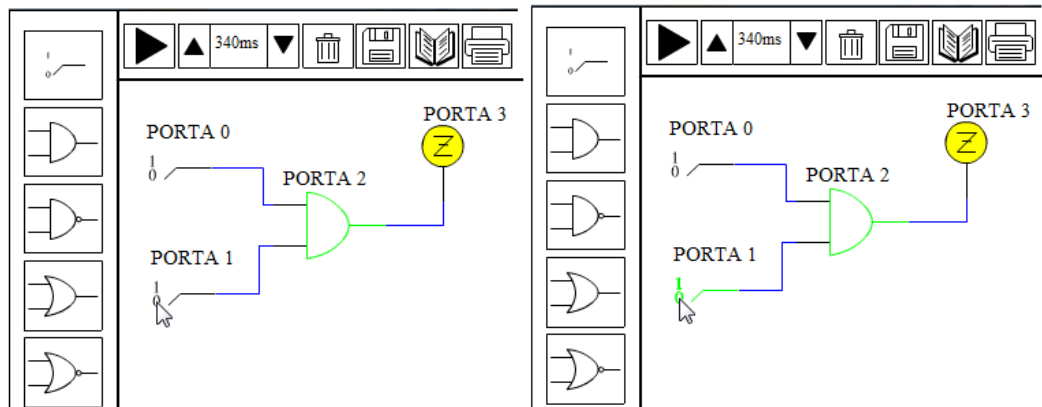


Figura 4.52 – Inserção da segunda porta lógica ao Grupo Selecionado.

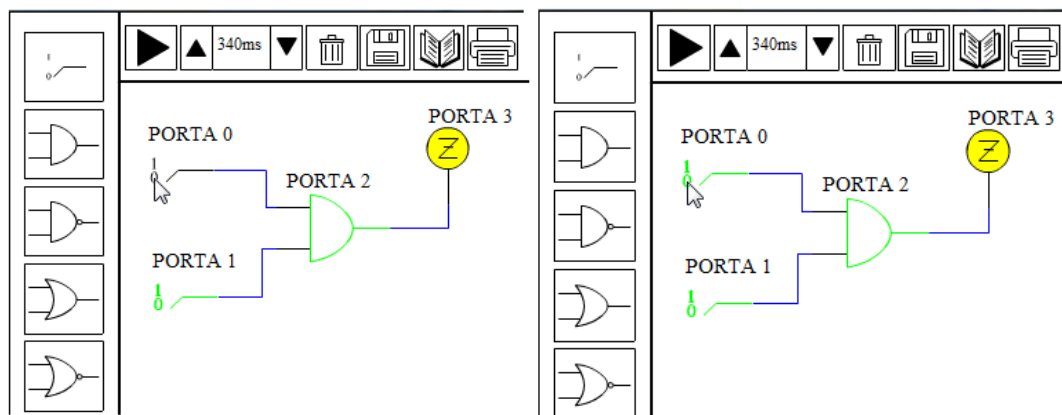


Figura 4.53 – Inserção da terceira porta lógica ao Grupo Selecionado.

Para excluir alguma porta lógica ou fio do grupo selecionado, deve-se repetir o processo de seleção sobre alguma porta lógica ou fio que já façam parte do Grupo Selecionado, ou seja, que apresentem cores que indicam estarem selecionados, mantendo a tecla “SHIFT” selecionada. Assim a porta lógica ou fio selecionado é excluído do Grupo Selecionado e retorna a sua cor original.

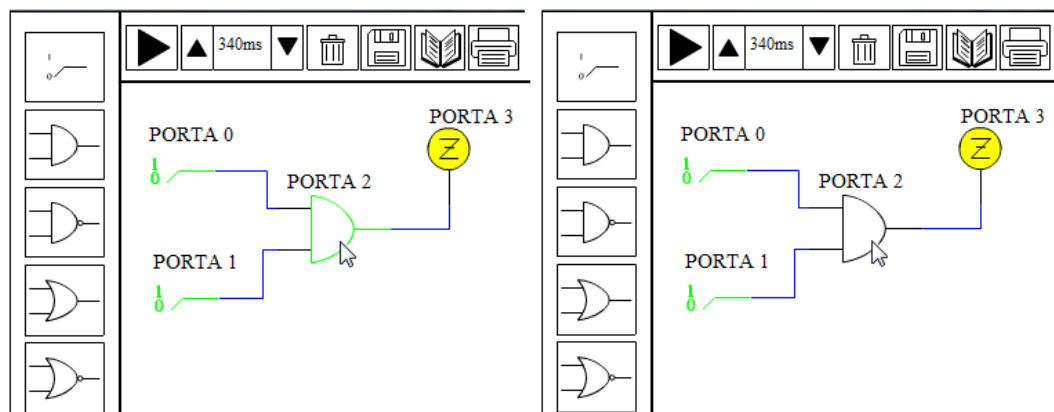


Figura 4.54 – Exclusão de uma porta lógica do Grupo Selecionado.

A seleção de um grupo de portas lógicas ou fios pode ser feita pela seleção individual de cada um dos componentes, ou através de um método que utiliza uma Caixa de Seleção. Para ativar o método de seleção utilizando a Caixa de Seleção deve-se posicionar o cursor do mouse sobre a Área de Trabalho, porém sem estar sobre uma porta lógica ou fio, e pressionar e manter pressionado o botão direito do mouse. Com o botão pressionado, ao mover o cursor do mouse a Caixa de Seleção será criada e acompanhará o cursor do mouse. Ao soltar o botão direito do mouse, as portas lógicas que estiverem em contato com a Caixa de Seleção entrarão para o Grupo Seleccionado e tem sua cor modificada. Os fios que tiverem suas duas pontas dentro da Caixa de Seleção também são selecionados.

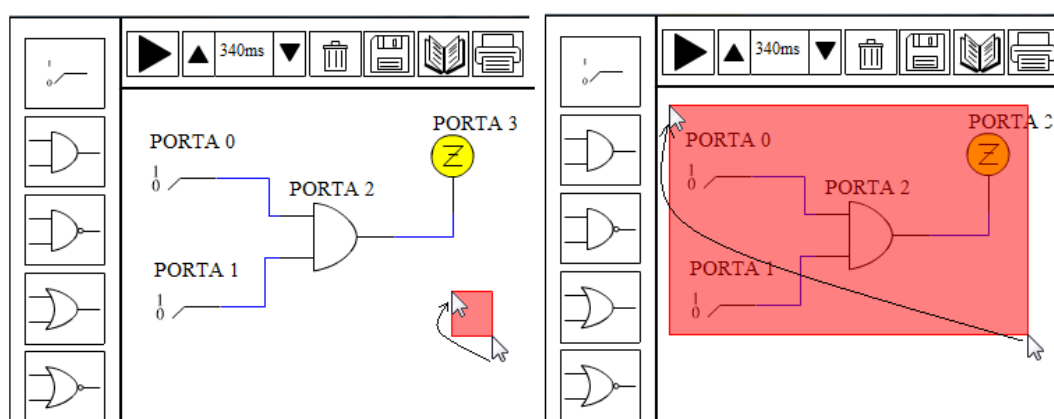


Figura 4.55 – Seleção de um conjunto de portas lógicas e fios utilizando a Caixa de Seleção.

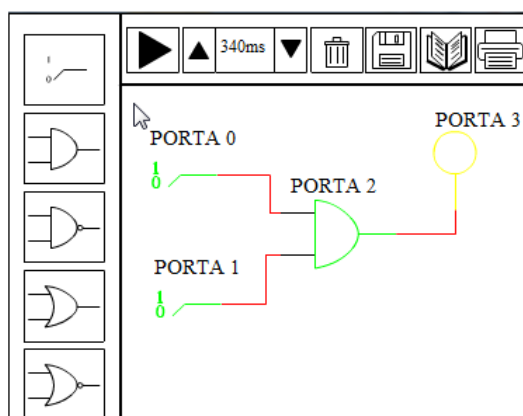


Figura 4.56 – Final do processo de seleção de todas as portas lógicas e fios do circuito utilizando a caixa de seleção.

É possível utilizar o processo de seleção utilizando Caixa de Seleção para inserir novas portas lógicas e fios a um grupo que contenha portas lógicas e fios previamente selecionados. Deve-se realizar o processo descrito para seleção utilizando Caixa de Seleção, porém mantendo a tecla “SHIFT” pressionada. Assim todas as portas lógicas que estiverem em contato com a Caixa de Seleção e os fios que tiverem suas duas pontas dentro da Caixa de

Seleção, serão incluídas no Grupo Seleccionado e terão a cor modificada. Caso alguma porta lógica ou fio selecionado como descrito acima já fizer parte do Grupo Seleccionado nada ocorre com estes. Não há no método de seleção utilizando Caixa de Seleção, a possibilidade de retirar portas lógicas ou fios do Grupo Seleccionado, é necessário retirar as portas lógicas e fios um a um do grupo.

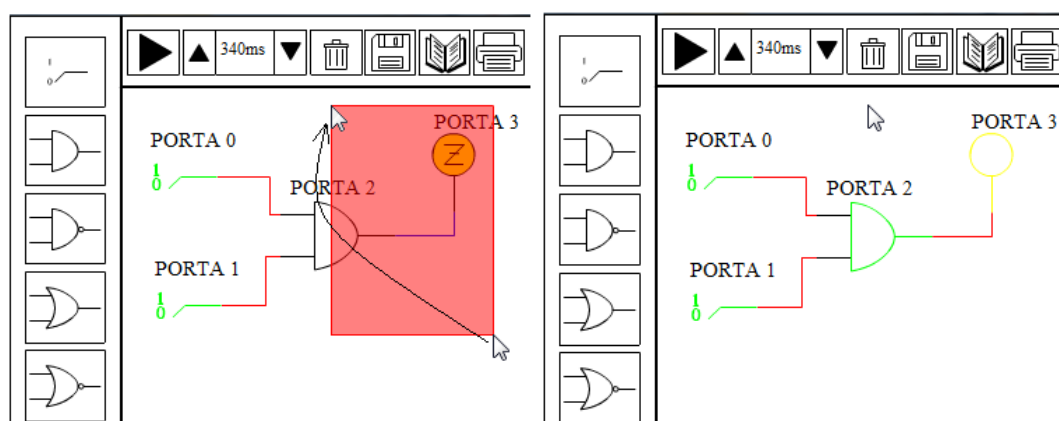


Figura 4.57 – Inserção de portas lógicas e fios ao Grupo Seleccionado que já continha outras portas lógicas e fios.

A inicialização do método de seleção utilizando a Caixa de Seleção, ou a ação de pressionar o botão direito do mouse, estando o cursor do mouse sobre a Área de Trabalho e não sobre uma porta lógica ou fio, ambos sem pressionar a tecla SHIFT, provoca a perda da informação do último Grupo Seleccionado. Todos os componentes do último grupo voltam a suas cores originais.

Existe integração do Grupo Seleccionado com algumas funções do programa para que elas possam ser aplicadas a todo o Grupo Seleccionado de uma só vez, não sendo necessário aplicar a função desejada a cada um dos componentes do grupo individualmente. As funções que possuem integração com a seleção de grupos são de: arrasto, eliminação, duplicação e rotação.

4.5.7. Arrasto

A função de arrasto é dividida em dois modos, manual e automático. O modo manual é realizado pelo usuário e para ser ativado deve-se selecionar o botão **ARRASTAR**, o que resulta na mudança da imagem exibida para uma que indique que a função se mantém ativa. Essa função não poder ser ativada quanto se estiver adicionando portas lógicas ou criando fios

na Área de Trabalho, quando a tabela do Editor de Minimização estiver inserida na Área de Trabalho, quando o Editor de Interpretação de Funções estiver ativo, ou quando as funções impressão, salvamento e carregamento de arquivos tiverem chamado as janelas do sistema operacional.

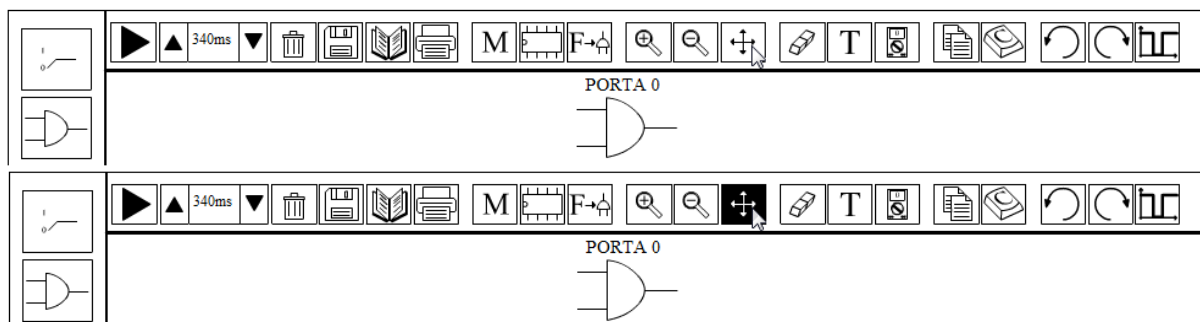


Figura 4.58 – Ativação da função de arrasto.

Com a função ativa, dois tipos de arrasto são possíveis, o arrasto da Área de Trabalho por completo e o arrasto de grupos de portas lógicas e fios selecionados. Para realizar o arrasto da Área de Trabalho deve-se posicionar o cursor do mouse sobre a Área de Trabalho, porém não sobre uma porta lógica ou fio, e em seguida pressionar e manter pressionado o botão esquerdo do mouse. Ao movimentar o cursor do mouse com o botão pressionado, a Área de Trabalho irá acompanhar esse movimento.

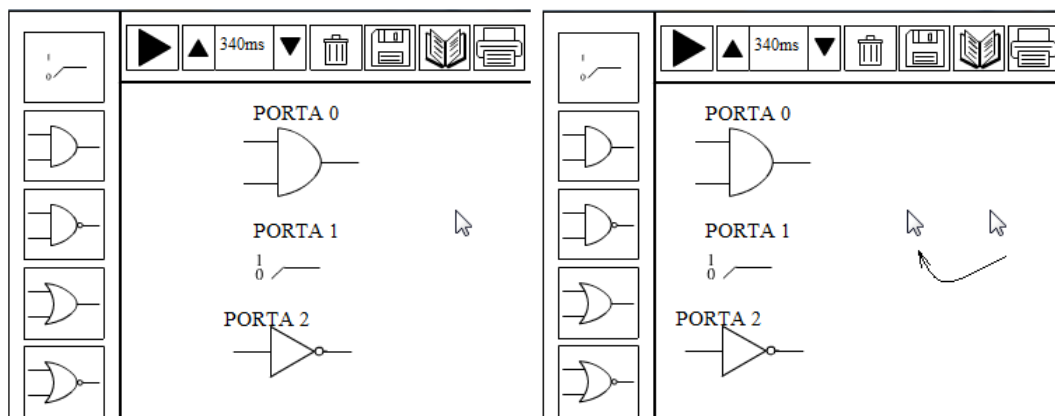


Figura 4.59 – Arrasto da Área de Trabalho.

Para realizar o arrasto de um grupo de portas lógicas e fios previamente selecionados, deve-se repetir o procedimento anterior, porém com a diferença que se deve posicionar o cursor do mouse sobre algum componente do Grupo Selecionado, assim somente o Grupo Selecionado será arrastado, deixando tanto a Área de Trabalho quanto as demais portas lógicas e fios, que não estão selecionados, na mesma posição.

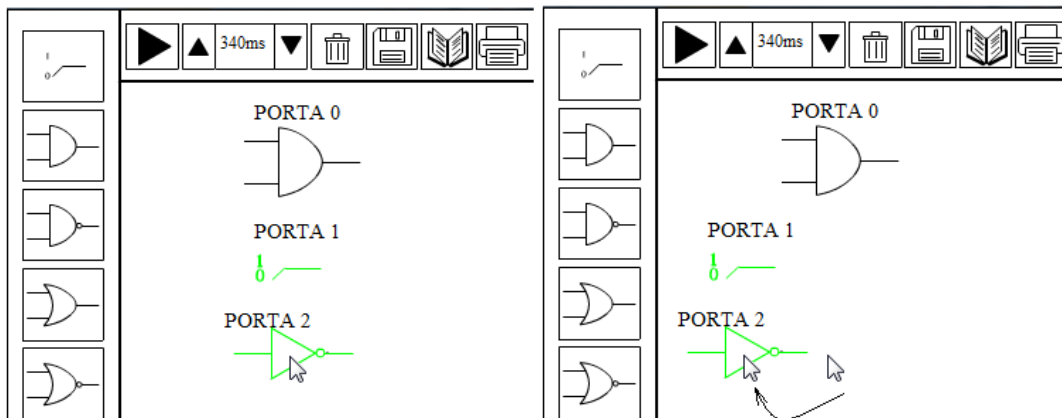


Figura 4.60 – Arrasto do Grupo Selecionado.

Para interromper o arrasto, deve-se soltar o botão esquerdo do mouse, assim a posição da Área de Trabalho ou do Grupo Selecionado no momento em que se solta o botão será mantida.

O arrasto automático ocorre quando algum componente está sendo inserido sobre a Área de Trabalho, como no caso da adição de portas lógicas, criação de fios, duplicação de componentes e finalização do Editor de Interpretação de Funções. O arrasto do Grupo Selecionado também ativa o arrasto automático. O arrasto automático é aplicado pelo programa toda vez que, em uma das situações descritas acima, o cursor do mouse se encontrar próximo a uma das arestas da janela do programa. O arrasto automático é aplicado sobre a Área de Trabalho e será sempre realizado na direção contrária a posição do cursor do mouse. Se durante a inserção de uma determinada porta lógica na Área de Trabalho, o cursor do mouse for posicionado próximo a aresta superior da janela do programa, a Área de Trabalho será arrastada para baixo. Assim é possível alcançar regiões da Área de Trabalho que no momento não são visíveis na janela do programa, eliminando a necessidade de realizar o arrasto manual da Área de Trabalho, antes de uma das ações mencionadas acima.

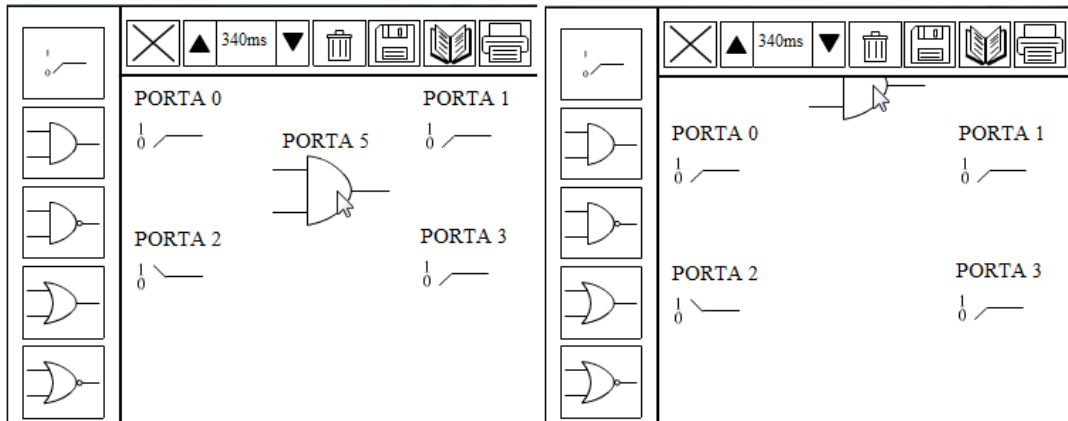


Figura 4.61 – Arrasto automático da Área de Trabalho durante a criação de uma porta lógica.

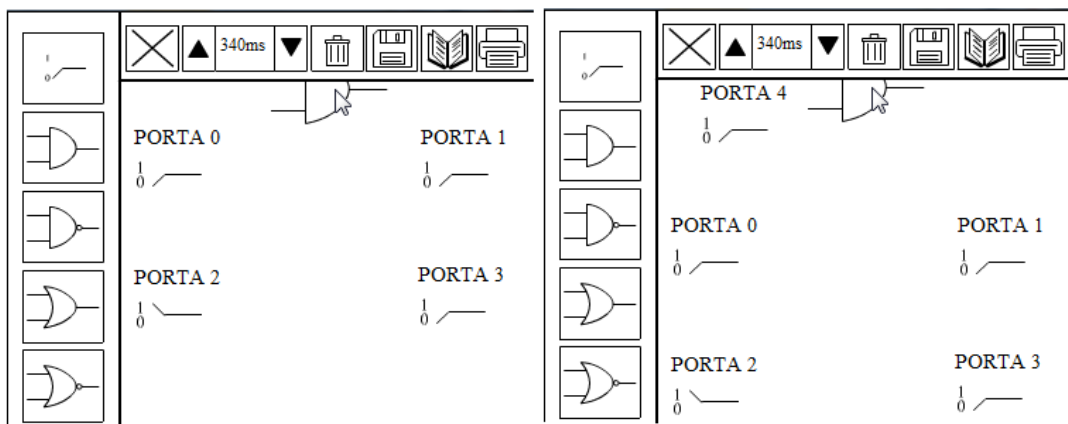


Figura 4.62 – Arrasto automático da Área de Trabalho revela portas lógicas não visíveis anteriormente.

O arrasto automático da Área de Trabalho pode ocorrer em duas direções, desde que o cursor do mouse esteja próximo a duas arestas adjacentes da janela do programa simultaneamente.

A Área de Trabalho tem a forma de um quadrado de tamanho máximo de 6000 pixels por 6000 pixels, podendo chegar a um tamanho mínimo de 2000 pixels por 2000 pixels. Uma vez que a janela do programa pode ser redimensionada a um tamanho mínimo de 1000 pixels por 600 pixels e que não há valores máximos para suas dimensões, algumas situações podem ocorrer. A primeira delas é que a Área de Trabalho tenha dimensões superiores às dimensões da janela do programa, nesse caso apenas uma porção da Área de Trabalho pode ser exibida na janela do programa e é necessário realizar o arrasto da Área de Trabalho para poder visualizar as demais partes desta. Ainda nesse caso é necessário observar alguns limites para essa operação de arrasto, que são os seguintes:

1. A aresta superior da Área de Trabalho nunca pode estar abaixo da aresta inferior da Barra de Ferramentas;
2. A aresta inferior da Área de Trabalho nunca pode estar acima da aresta superior da Barra de Informações;
3. A aresta esquerda da Área de Trabalho nunca pode estar à direita da aresta direita do Container de Portas;
4. A aresta direita da Área de Trabalho nunca pode estar à esquerda da aresta direita da janela do programa;

Essas regras são aplicadas automaticamente pelo programa, que impede o arrasto em alguma direção caso esse arrasto venha a desobedecer algumas das regras acima. Caso o arrasto seja feito em mais de uma direção, apenas a direção em que o arrasto for desobedecer às regras acima é bloqueado, o arrasto na outra direção continua liberado.

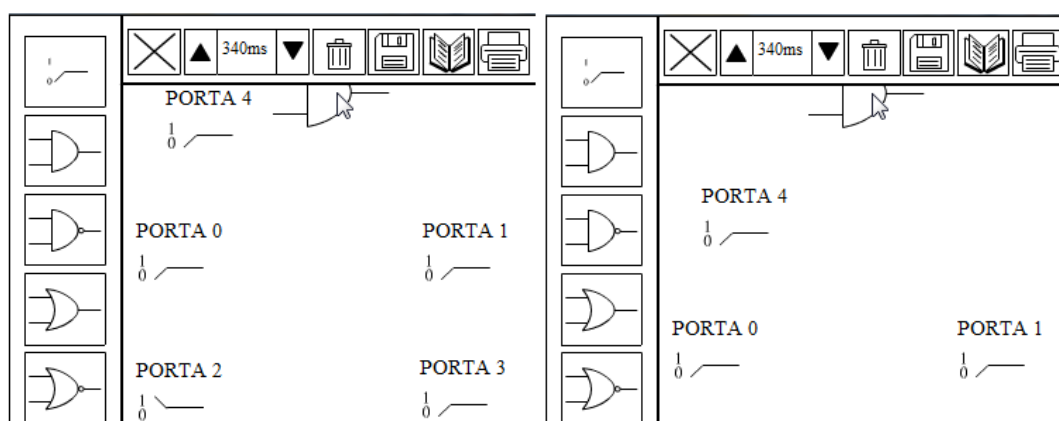


Figura 4.63 – Arrasto automático da Área de Trabalho é bloqueado quando o limite de posicionamento for alcançado.

A segunda situação é quando a janela do programa possui dimensões superiores às dimensões da Área de Trabalho. Pode ocorrer da Área de Trabalho ter apenas uma das dimensões inferior à dimensão da janela do programa, nesse caso apenas a dimensão que for inferior terá o arrasto bloqueado, uma vez que é possível visualizar toda a extensão da Área de Trabalho nessa dimensão dentro da janela do programa. Também pode ocorrer de as duas dimensões serem inferiores às dimensões da janela do programa, logo o arrasto fica bloqueado nessas duas dimensões. Em todas essas situações uma região amarela é exibida preenchendo o espaço entre a Área de Trabalho e a janela do programa.



Figura 4.64 – Área de Trabalho menor que a janela do programa horizontalmente, bloqueio do arrasto nesta direção.

O arrasto do Grupo Seleccionado respeita limites, assim como no caso do arrasto da Área de Trabalho. Nesse caso o grupo arrastado só acompanha o movimento do cursor do mouse enquanto a área do grupo não ultrapassa as arestas da Área de Trabalho. Assim o programa impede que o grupo arrastado ultrapasse esses limites bloqueando o arrasto somente na direção que poderia ocasionar a situação de ultrapassagem.

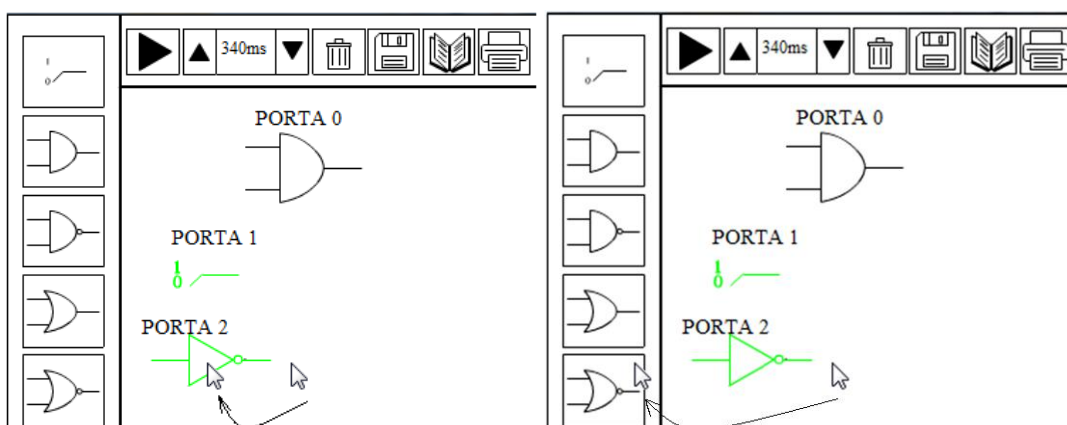


Figura 4.65 – Bloqueio do arrasto do Grupo Seleccionado ao encontrar um limite da Área de Trabalho.

A função de arrasto possui integração com algumas funções do programa: Deletar, Editor de Minimização, Editor de Macros, Eliminação, Seleção, Impressão e Simulação. Somente quando nenhuma função ou editor encontra-se ativo, ou durante a execução de uma das funções citadas acima é possível ativar a função de arrasto, porém ela encontra-se limitada ao arrasto da Área de Trabalho, menos no Editor de Macros onde não há limitação.

Para finalizar a função de arrasto deve-se seleccionar o botão **ARRASTAR**, assim a imagem original volta a ser exibida e a função é desativada. Também há a possibilidade de se ativar a função de ampliação ou redução para desativar automaticamente a função de arrasto, devido ao funcionamento dos três botões dessas funções como botões de rádio.

Ao se realizar o arrasto individual de portas lógicas que possuam fios ligados a seus pinos de entrada ou saída, esses fios terão suas propriedades alteradas para manter a informação visual da ligação coerente. Com o arrasto da porta lógica, as arestas do fio podem ser arrastadas ou redimensionadas para se manter junto ao pino da porta lógica. Dessa maneira o fio continua representando visualmente a ligação que existia antes do arrasto da porta lógica. Esse arrasto e redimensionamento terminam quando o arrasto da porta lógica termina.

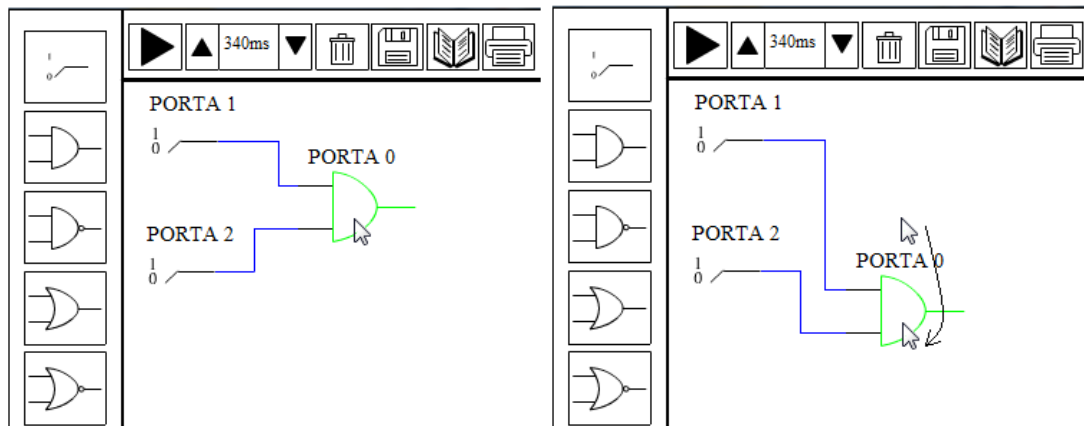


Figura 4.66 – Arrasto da porta lógica provoca o redimensionamento dos fios ligados a seus pinos.

Caso existam fios ligados à aresta editada no processo acima detalhado, esses fios também sofrem edição. O Ponto Indicador de Ligação entre os fios também tem sua posição editada para acompanhar a edição da posição dos fios. A finalidade da edição nesses fios é a mesma, manter coerente a informação visual da ligação entre os fios. Dessa maneira as arestas desse fio podem ser arrastadas ou redimensionadas. Admitindo-se haver inúmeras ligações no fio, o processo descrito acima será empregado em todas as situações em que haja necessidade de se manter a informação visual dessas ligações. O programa gerencia o emprego desse método para manter o circuito sempre em conformidade com o circuito anterior ao arrasto.

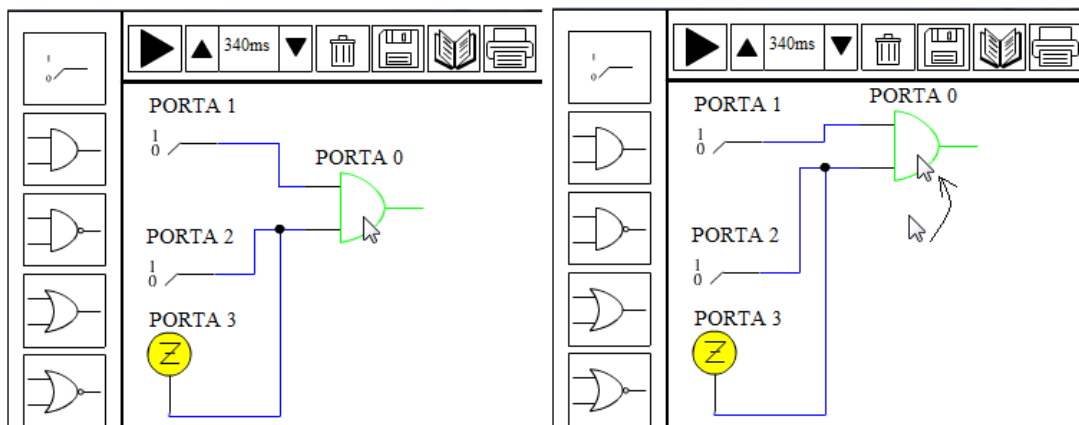


Figura 4.67 – Edição dos fios ligados à porta lógica arrastada provoca edição dos outros fios ligados a esses fios.

O processo descrito para o arrasto individual de portas lógicas também ocorre no caso de arrasto de um Grupo Selecionado, em que fios ligados a saídas e entradas de portas lógicas do grupo não estejam selecionadas. Nesse caso o processo aplicado sobre esses fios será o mesmo, porém cada fio acompanha a porta lógica com a qual possui ligação. No caso do Grupo Selecionado conter duas portas lógicas que são ligadas por um fio e esse fio não estiver selecionado, o fio será redimensionado de maneira a manter suas pontas sempre em contato com os pinos com os quais existe ligação.

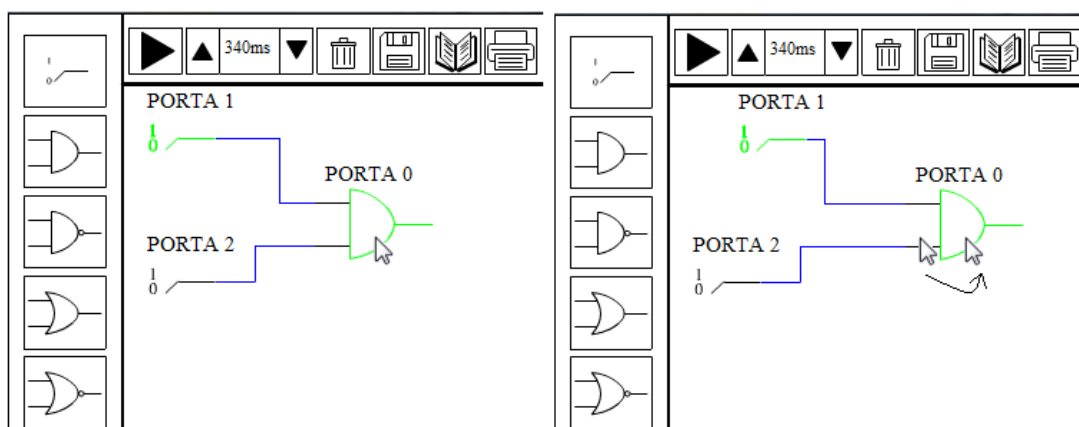


Figura 4.68 – Edição das posições das pontas de um fio para acompanhar o arrasto do Grupo Selecionado.

Para que um fio seja arrastado por completo e não redimensionado, as duas portas lógicas às quais este fio faz a ligação devem estar selecionadas e o próprio fio deve estar selecionado. Assim o arrasto de qualquer elemento do grupo selecionado vai realizar o arrasto do fio.

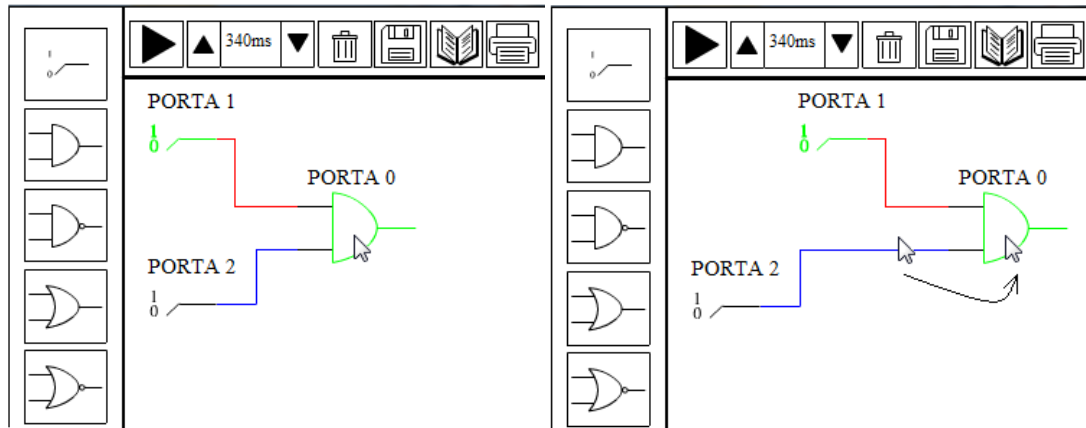


Figura 4.69 – Fio selecionado é arrasto por completo.

Caso o Grupo Selecionado alvo do processo de arrasto seja composto somente por um fio, ao se iniciar o arrasto deste fio, o arrasto não será realizado no fio por completo, e sim na aresta sobre a qual o cursor do mouse está posicionado. Assim somente a aresta em questão, e as arestas adjacentes, serão afetadas e todo o restante do fio permanecerá o mesmo. Nesse caso a aresta selecionada irá mudar de posição com o movimento do cursor do mouse, e as arestas adjacentes a essa aresta serão redimensionadas para acompanhar o movimento da aresta. O arrasto dessa aresta possui limitações, caso a aresta possua orientação vertical, seu arrasto somente poderá ser feito horizontalmente, e caso a aresta possua orientação horizontal, seu arrasto somente poderá ser feito verticalmente.

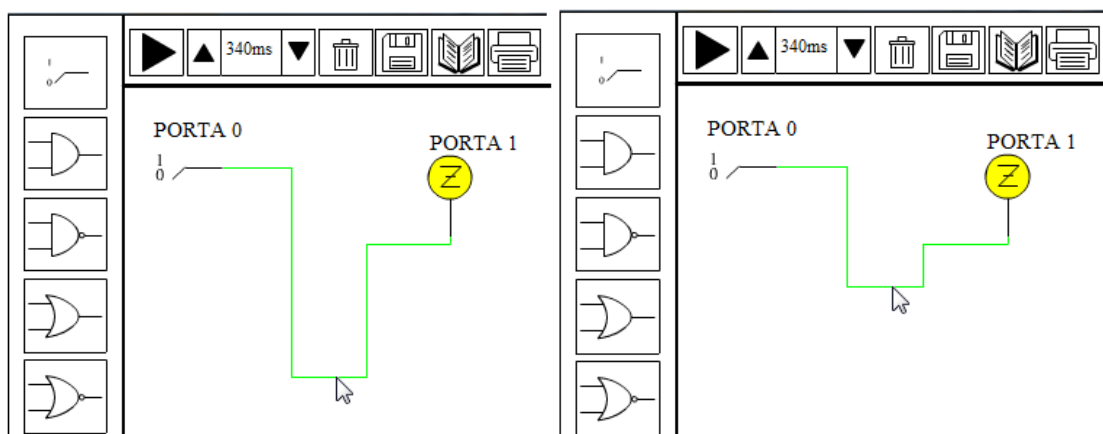


Figura 4.70 – Arrasto da aresta selecionada provoca o redimensionamento das arestas adjacentes.

Caso haja fios ligados às arestas editadas pelo processo de arrasto, o programa se encarregará de atuar sobre esses fios editando suas arestas para que a informação visual da ligação entre eles se mantenham coerentes com o circuito anterior ao arrasto.

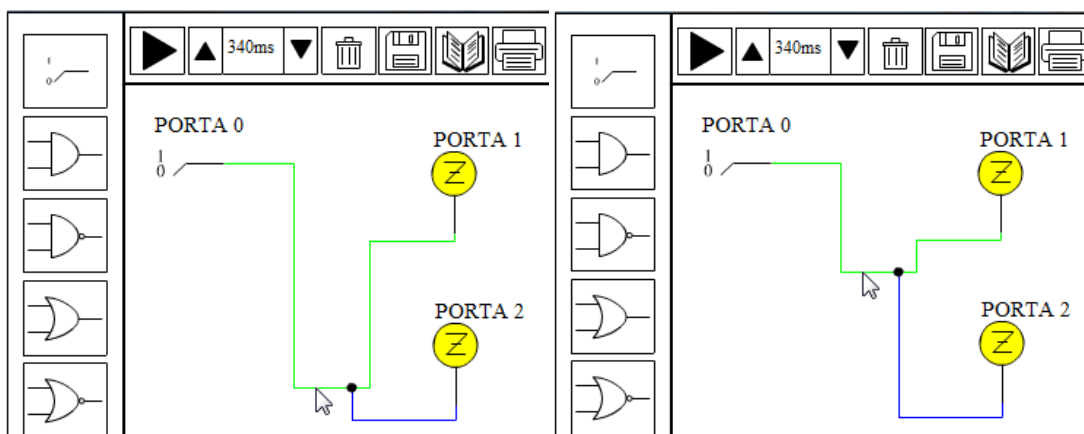


Figura 4.71 – Arrasto da aresta do fio selecionado provoca o redimensionamento do fio ligado a essa aresta.

Existe um caso especial de arrasto de aresta de um fio, assim como ilustrado na Figura 4.71, em que existe um fio ligado a uma das arestas adjacentes à aresta arrastada. Nesse caso o processo de arrasto provoca o redimensionamento das arestas adjacentes, porém o arrasto não provoca a edição do último fio, o que ocorre é uma situação de "escorregamento" entre os fios, em que o fio e seu Ponto Indicador de Ligação permanecem inalterados enquanto a aresta é redimensionada. Essa situação de "escorregamento" se mantém até o momento em que o Ponto Indicador de Ligação encontra uma das quinas da aresta e nesse momento o fio passa a ser redimensionado e o Ponto Indicador de Ligação tem sua posição modificada passando a acompanhar a posição da quina.

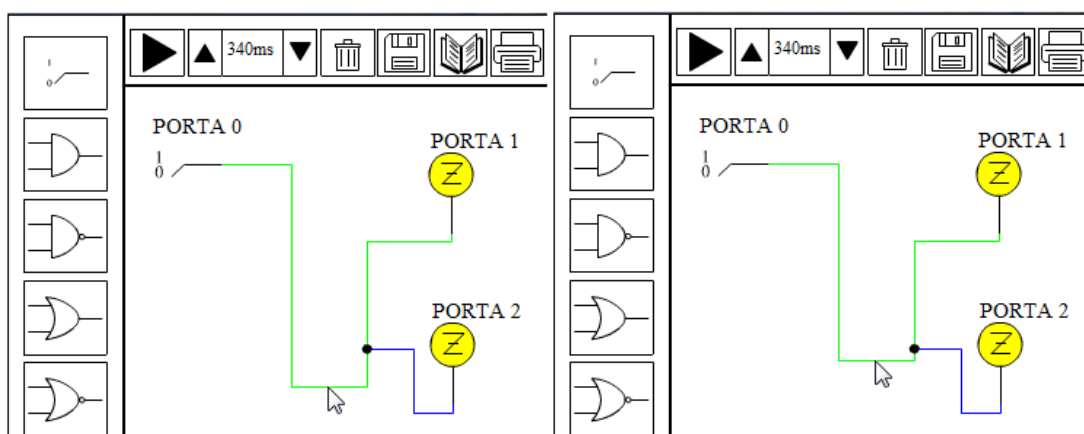


Figura 4.72 – Na situação de “escorregamento” o arrasto da aresta não edita o fio ligado à aresta adjacente.

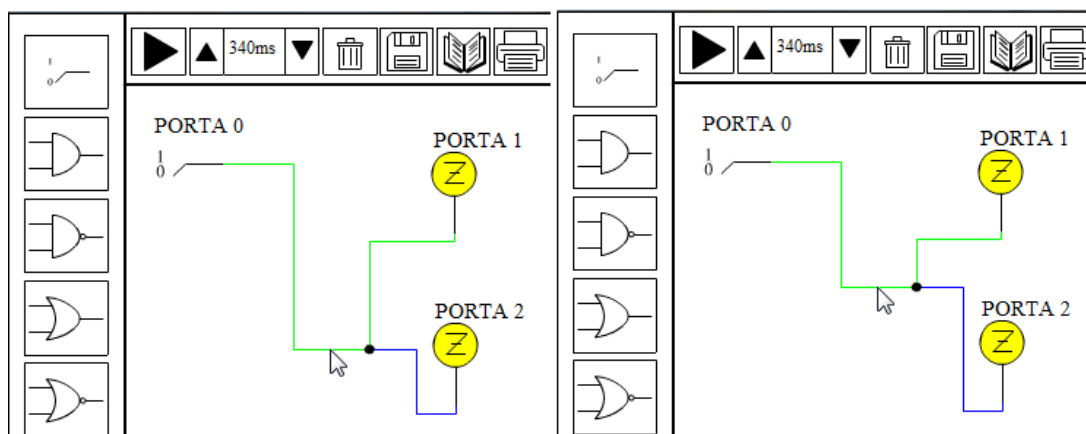


Figura 4.73 – Fora da situação de “escorregamento” o arrasto da aresta edita o fio ligado a aresta adjacente.

O processo ilustrado na Figura 4.73 é encerrado caso o arrasto da aresta seja feito no sentido contrário ao que era feito, assim o Ponto Indicador de Ligação deixa de acompanhar o movimento da aresta e permanece na posição em que estava no momento em que o arrasto mudou de sentido. A Figura 4.74 ilustra essa situação de retorno do “escorregamento”.

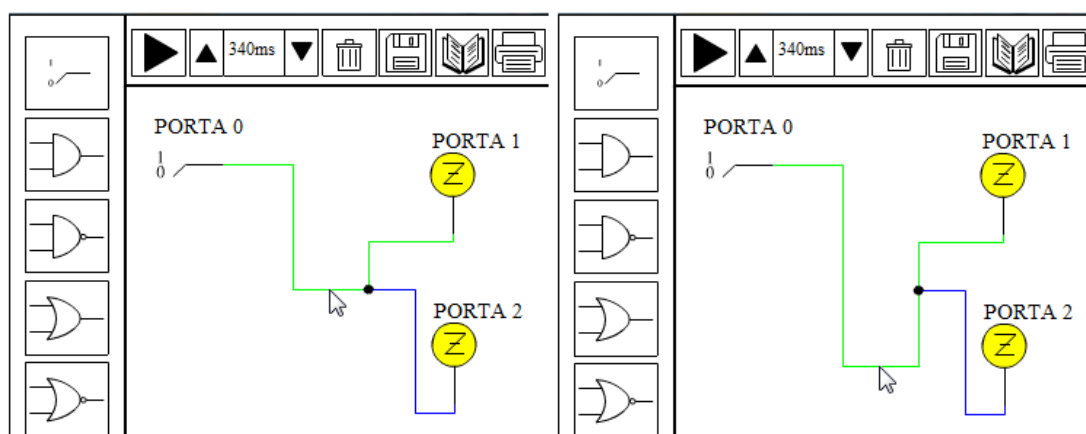


Figura 4.74 – Situação de “escorregamento” volta a ocorrer quando o arrasto da aresta é feito em sentido contrário.

Ainda no processo de arrasto ilustrado na Figura 4.73, pode existir uma situação especial em que o redimensionamento da aresta adjacente pode provocar a inversão das quinas dessa aresta. Nesse caso a quina arrastada que estava abaixo da outra quina passa a ficar posicionada acima da outra quina. Essa ultrapassagem faz com que o Ponto Indicador de Ligação permaneça junto à quina que não teve a posição editada. O Ponto Indicador de Ligação permanece sobre a quina que não teve sua posição editada mesmo em uma nova situação de inversão das quinas da aresta adjacente.

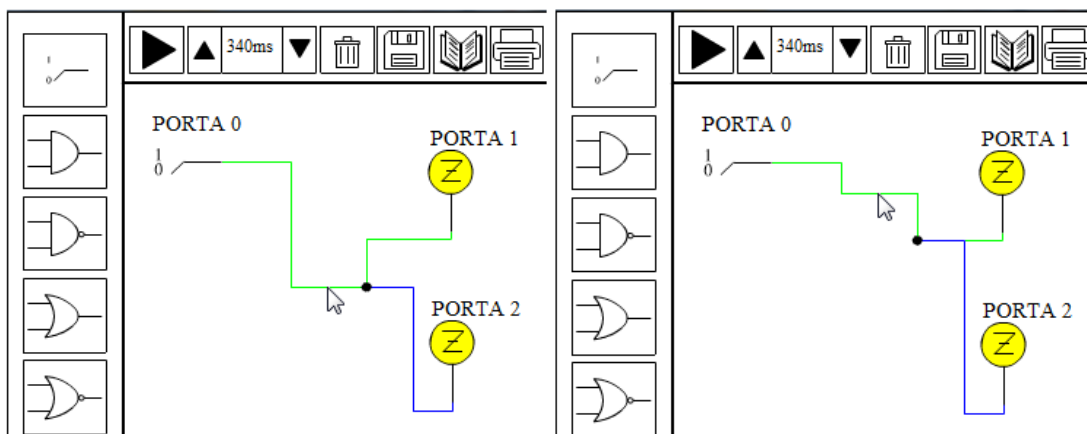


Figura 4.75 – Situação de inversão das duas quinas da aresta adjacente à aresta arrastada.

Caso a aresta selecionada para o arrasto seja uma das arestas das extremidades do fio, o processo de arrasto, além de editar a posição dessa aresta e redimensionar a aresta adjacente, irá adicionar uma nova aresta ao fio, caso essa extremidade esteja ligada a um pino de uma porta lógica, ou provocará uma situação de "escorregamento" entre fios, caso essa extremidade esteja ligada a outro fio. No caso da adição de uma nova aresta, essa nova aresta irá completar a ligação do fio com o pino da porta lógica com a qual o fio é ligado e terá orientação contrária a da aresta arrastada. A nova aresta também será redimensionada para acompanhar o movimento da aresta arrastada. Ao fim do arrasto da aresta, a nova aresta estará inserida no fio.

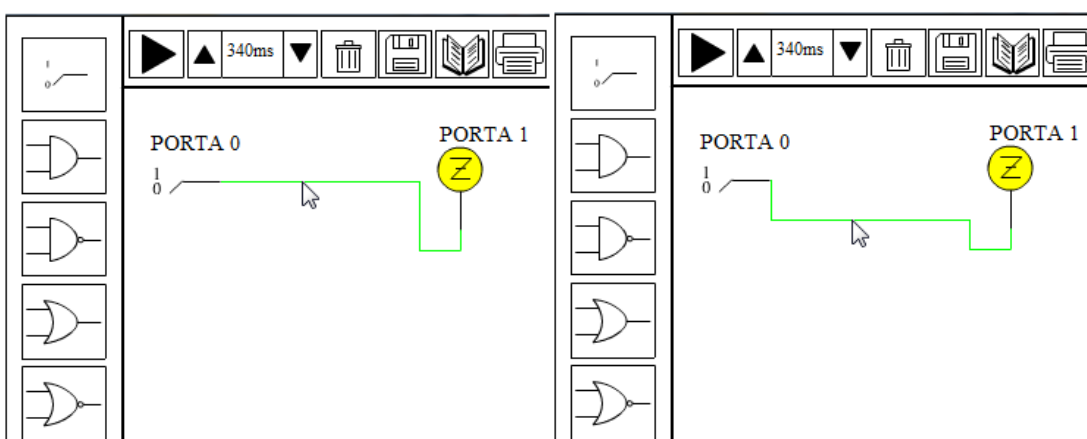


Figura 4.76 – Processo de arrasto da aresta provoca a criação de uma nova aresta no fio.

Caso a aresta de extremidade arrastada esteja ligada a outro fio, a situação de "escorregamento" entre os fios ocorre. A aresta arrastada tem sua posição editada conforme o movimento do cursor do mouse e junto à aresta o Ponto Indicador de Ligação também tem sua posição editada. Porém o arrasto respeita os limites da aresta com a qual o Ponto Indicador de Ligação faz ligação. Assim o Ponto Indicador de Ligação acompanha o

movimento do cursor do mouse somente enquanto não encontra uma das quinas da aresta com a qual faz ligação, nesse momento ele permanece junto à quina em questão, ignorando o movimento do cursor do mouse enquanto esse o forçar a ultrapassar a quina. O Ponto Indicador de Ligação voltará a acompanhar o movimento do cursor do mouse, somente quando esse movimento for realizado em sentido contrário, ou seja, quando parar de forçar a posição do Ponto Indicador de Ligação além da quina. Dessa maneira, o cursor do mouse não estará mais posicionado sobre a aresta arrastada do fio, porém ela ainda será arrastada.

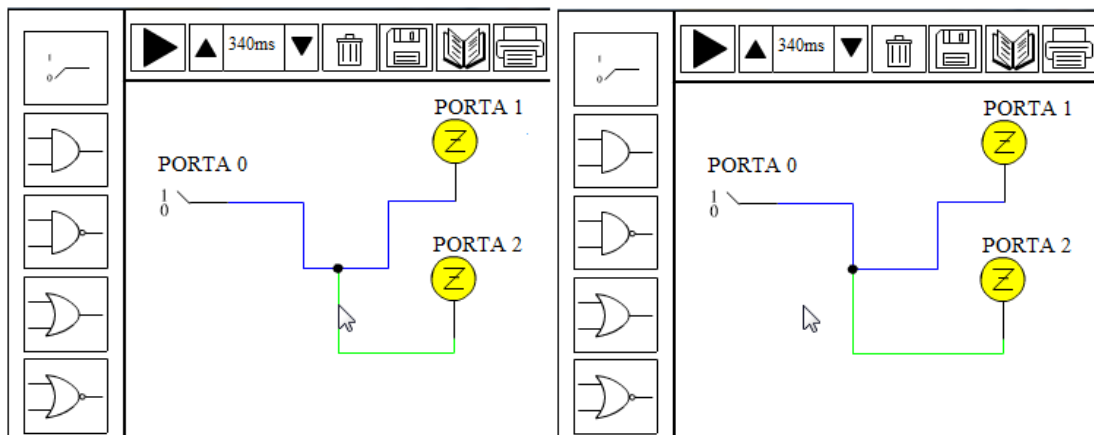


Figura 4.77 – Bloqueio do “escorregamento” do Ponto Indicador de Ligação durante o arrasto da aresta.

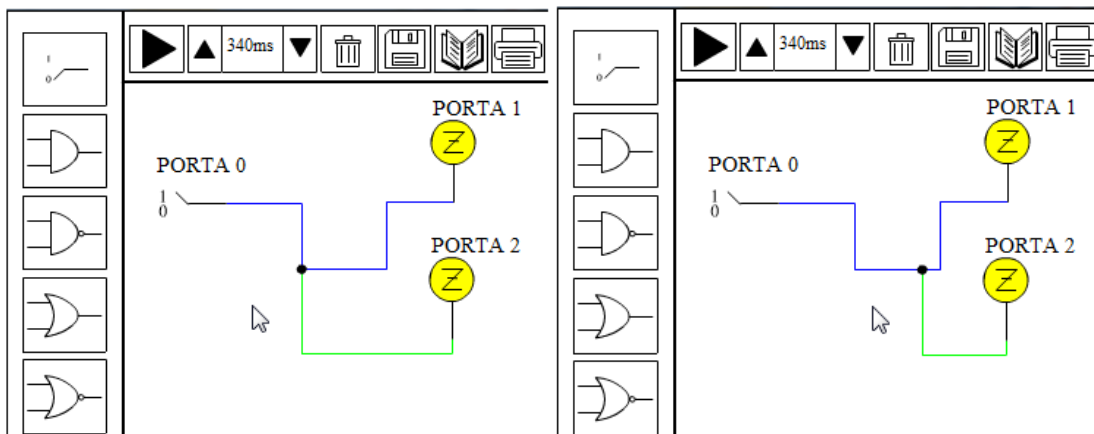


Figura 4.78 – Retomada do “escorregamento” do Ponto Indicador de Ligação para arrasto em sentido contrário.

Se o processo de arrasto for encerrado após o processo ilustrado na Figura 4.77, onde o Ponto Indicador de Ligação encontra-se sobre uma das quinas do fio com o qual faz ligação, pode ocorrer uma situação de Mudança de Aresta de Ligação. Nessa situação o Ponto Indicador de Ligação deixa de estar ligado à aresta com a qual estava antes do início do processo de arrasto e passa a estar ligado com a aresta adjacente que divide a quina em que o

Ponto Indicador de Ligação está posicionado e assim em um futuro processo de arrasto esse Ponto Indicador de Ligação será arrastado sobre essa nova aresta. Essa situação só ocorre para arrasto em um determinado sentido, assim na Figura 4.77 se a aresta tivesse sido arrastada para a direita e não para esquerda, quando o processo de arrasto fosse encerrado, estando o Ponto Indicador de Ligação sobre a quina mais a direita, a situação de mudança de aresta de ligação não teria ocorrido. A determinação de qual sentido permite a ocorrência da situação de mudança de aresta de ligação, depende de uma série de fatores ligados somente à construção dos fios envolvidos nessa situação.

Segundo o que foi explicado acima, a situação de Mudança de Aresta de Ligação provoca a existência de um tipo de ligação entre fios que apresenta uma característica especial. A ligação em questão é aquela em que os fios que fazem parte desta, têm as arestas que estão em contato com o Ponto Indicador de Ligação, com a mesma orientação. Esse tipo de ligação também pode ser feita através do processo de criação de fios.

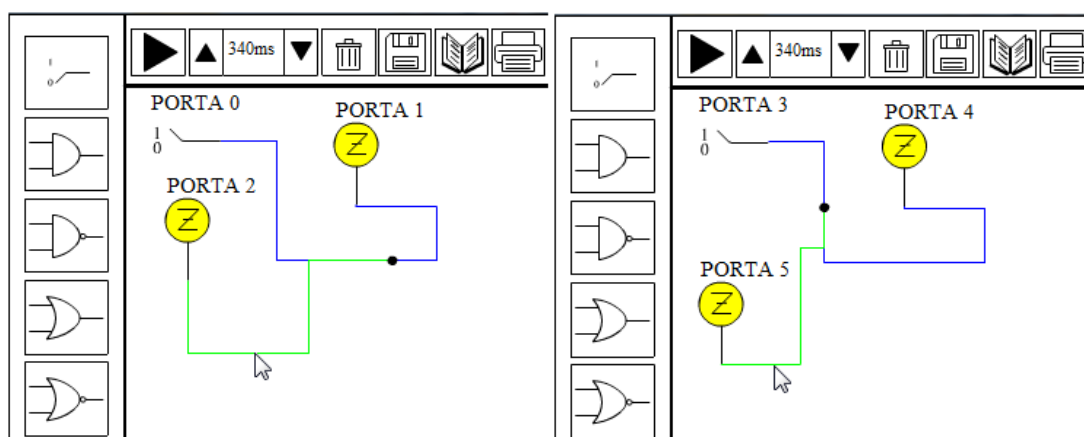


Figura 4.79 – Ligação especial em que as arestas em contato com Ponto Indicador de Ligação têm mesma orientação.

As ligações apresentadas na Figura 4.79 possuem uma diferença quando comparada as demais. O fio em que a quina da aresta da extremidade tem contato com o Ponto Indicador de Ligação, os fios que possuem cor verde claro na Figura 4.79, se comportam de maneira diferente quando a aresta dessa extremidade é arrastada. Quando essas arestas são arrastadas, ocorre um processo semelhante ao caso de arrasto de arestas de extremidade com ligações em pinos de portas lógicas. Nesse caso o arrasto provoca a criação de uma nova aresta no fio, que é inserida entre a quina arrastada e o Ponto Indicador de Ligação. Essa nova aresta tem orientação contrária à aresta arrastada e é redimensionada com o arrasto desta. A adição da nova aresta com orientação contrária a aresta arrastada, faz com que a ligação deixe de ser

caracteriza como ligação especial, logo passa a ser tratada pelo programa como as ligações normais.

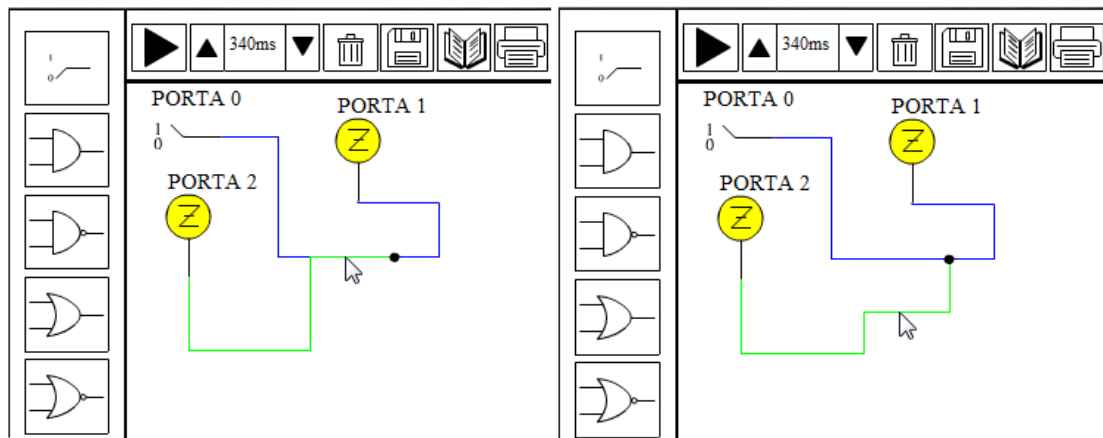


Figura 4.80 – Criação de uma nova aresta no fio que representava ligação especial.

Ao final do arrasto de uma aresta do fio, pode ocorrer uma situação em que duas quinas de um fio encontrem-se muito próximas uma da outra. Nesse caso o programa atuará sobre o fio eliminando as duas quinas e consequentemente a aresta entre elas. Porém, ao eliminar as duas quinas, as arestas adjacentes à aresta eliminada perdem cada uma, uma quina e assim também são eliminadas, logo o número de arestas eliminadas sobe para três. Após a eliminação das arestas, uma nova aresta é criada, contendo as quinas que sobram das arestas eliminadas. Assim esse processo pode ser utilizado para remover arestas indesejadas no fio. Porém existe uma restrição nesse processo de eliminação de arestas, como não é permitida a existência de fios com menos de quatro quinas, caso o fio resultante do processo de eliminação tenha menos de quatro quinas, esse processo não é realizado.

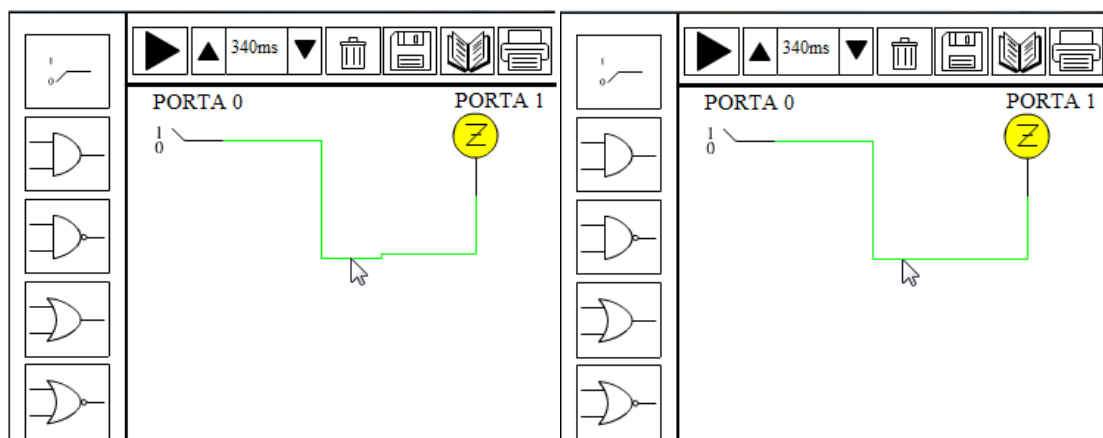


Figura 4.81 – Eliminação de aresta após finalização de arrasto.

Na Figura 4.81, é possível observar que não há a necessidade de que as quinas estejam exatamente com a mesma posição, basta que elas estejam suficientemente próximas, para que o processo de eliminação da aresta seja executado. Também é possível observar que após a eliminação das duas quinas, as quinas adjacentes a estas não possuem a mesma posição vertical, assim o programa se encarrega de reposicionar essas duas quinas antes de criar a aresta que utiliza essas duas quinas. No caso da Figura 4.81, a quina posicionada a direita das arestas eliminadas encontrava-se acima da quina posicionada a esquerda das arestas eliminadas. Assim a quina à direita teve sua posição vertical editada para corresponder à posição vertical da quina a esquerda. Após essa edição, a aresta criada utilizando essas duas quinas é inserida no fio.

Caso exista algum Ponto Indicador de ligação nas arestas eliminadas, ocorre a situação de Mudança de Aresta de Ligação, assim como foi descrito mais acima. O Ponto Indicador de Ligação passa a fazer ligação com a aresta que é criada após a eliminação.

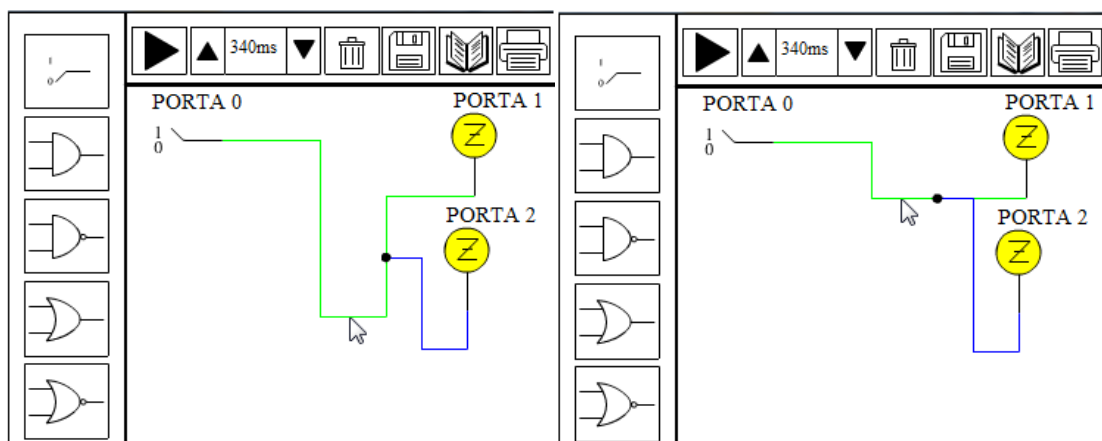


Figura 4.82 – Situação de Mudança de Aresta de Ligação do Ponto Indicador de Ligação do fio editado pelo arrasto.

O arrasto de uma aresta de um fio também respeita os limites da Área de Trabalho, ou seja, o arrasto da aresta é bloqueado para evitar que ela seja posicionada fora da Área de Trabalho.

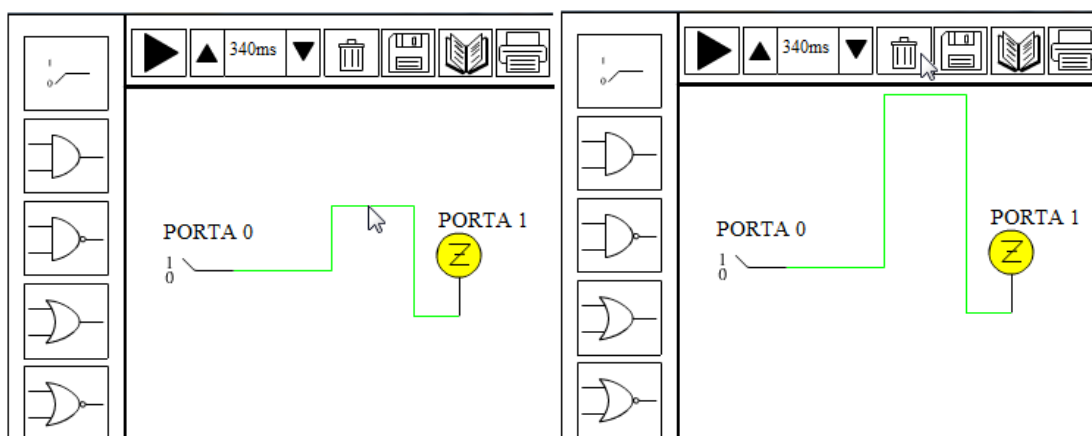


Figura 4.83 – Bloqueio do arrasto da aresta do fio quando o limite da Área de Trabalho é encontrado.

Caso duas portas lógicas ligadas às extremidades do fio estejam selecionadas assim como o próprio fio, o arrasto iniciado no fio irá realizar o arrasto de todo o grupo, e não só de uma aresta do fio, como foi apresentado na Figura 4.69. Porém, se alguma das portas ligadas às extremidades do fio não estiverem selecionadas, o arrasto iniciado pelo fio irá apenas realizar o arrasto dos demais componentes do Grupo Selecionado, assim o fio será redimensionado para manter a informação visual da sua ligação.

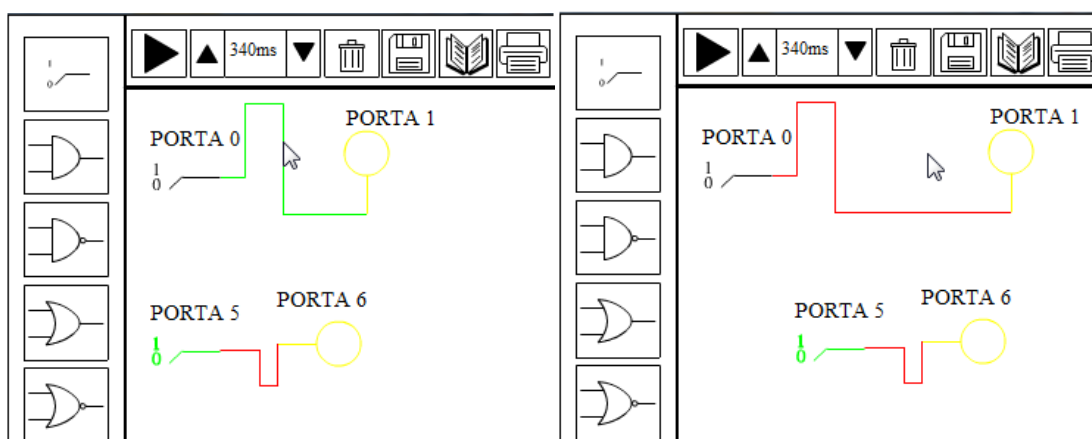


Figura 4.84 – Bloqueio do arrasto do fio, pois apenas uma das portas lógicas ligadas às pontas do fio está selecionada.

Na Figura 4.84 o fio por onde se inicia o processo de arrasto volta a ter a cor vermelha, pois o cursor do mouse não está mais sobre ele.

4.5.8. Ampliar / Reduzir

A função de ampliar e reduzir se aplica sobre a Área de Trabalho. Inicialmente com tamanho de 6000 pixels de largura por 6000 pixels de altura, a Área de Trabalho pode ser reduzida a um tamanho de 2000 pixels de largura por 2000 pixels de altura. Sempre que a Área de Trabalho é reduzida ou ampliada, todos os componentes sobre ela também são reduzidos ou ampliados na mesma proporção. Essa função permite visualizar toda a Área de Trabalho sem a necessidade de realizar um grande número de operações de arrasto sobre ela.

Para ativar essa função deve-se selecionar um dos botões Ampliar ou Reduzir, o que resulta na mudança da imagem do botão para uma imagem que indica que a função permanece ativa.

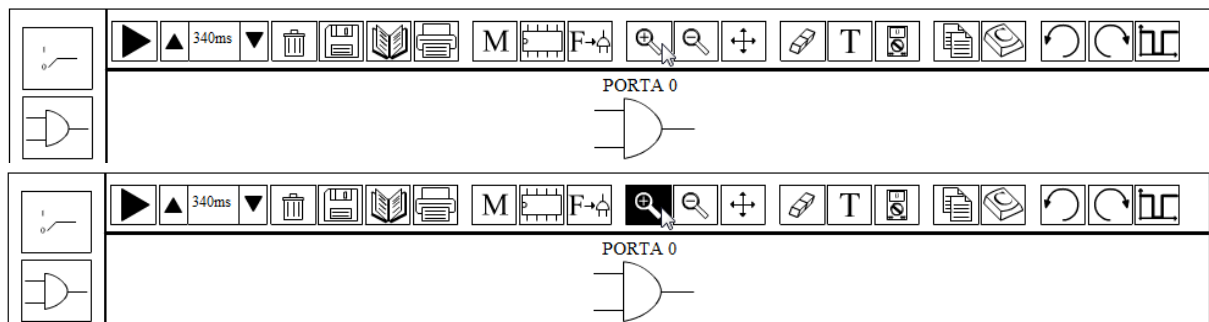


Figura 4.85 – Ativação da função Ampliar.

Para executar a função deve-se posicionar o cursor do mouse sobre a Área de Trabalho, inclusive sobre uma porta lógica ou fio, e pressionar o botão esquerdo do mouse. A redução ou ampliação da Área de Trabalho será feita tendo como pivô a posição onde o cursor do mouse se encontra no momento em que o botão é pressionado, e é sempre feita retirando ou adicionado 1/6 das dimensões originais da Área de Trabalho, ou seja, cada redução retira 1000 pixels em cada dimensão e cada ampliação adiciona 1000 pixels em cada direção.

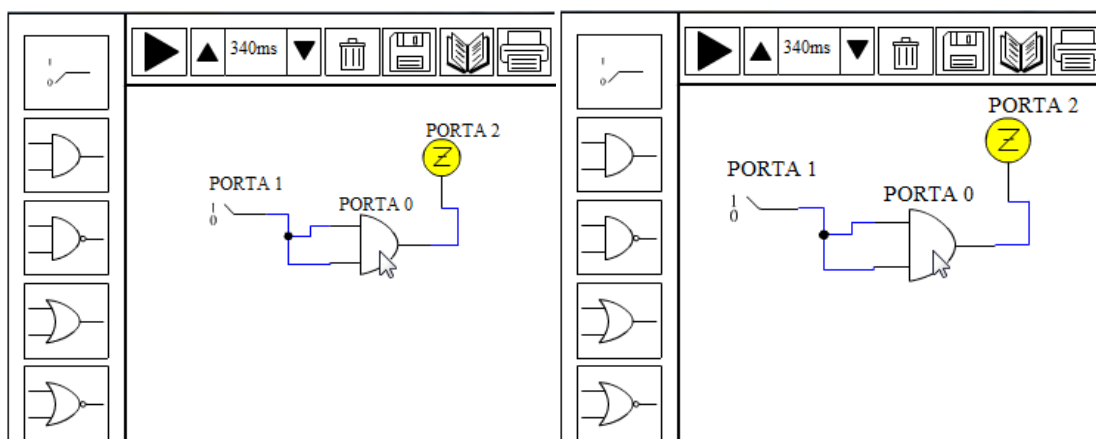


Figura 4.86 – Ampliação da Área de Trabalho.

O processo de ampliação sempre respeita a regra de manter o pivô na mesma posição do cursor do mouse, porém o processo de redução pode não respeitar em algumas situações. No caso do processo de redução ser aplicado muito próximo às extremidades da Área de Trabalho, se o pivô da redução fosse mantido na posição do cursor do mouse, poderia ocorrer de a Área de Trabalho desrespeitar os limites de posicionamento como foram descritos no capítulo 4.5.7 devido à redução de suas dimensões. Nesse caso o programa faz com que o pivô da redução seja aplicado em uma posição em que a posição da Área de Trabalho resultante não desrespeite os limites de posicionamento.

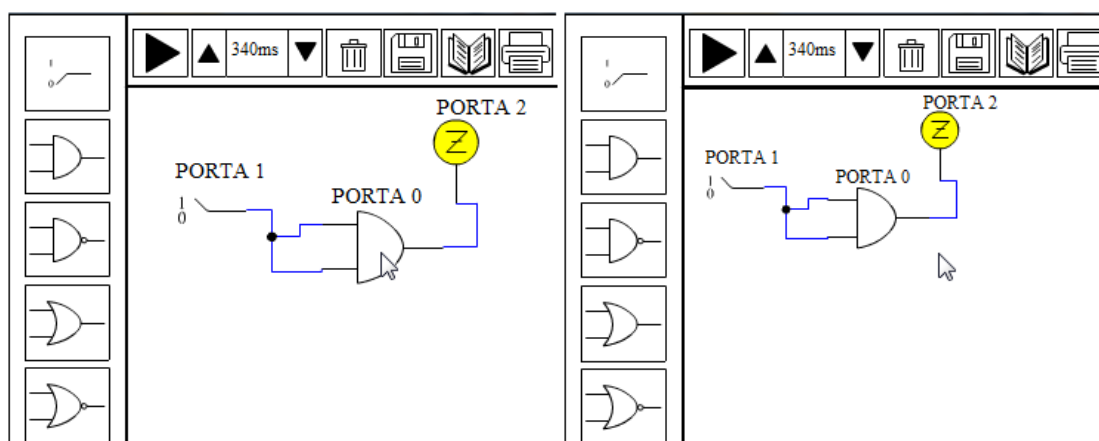


Figura 4.87 – Redução da Área de Trabalho com mudança de pivô.

A função de ampliação e redução possui integração com algumas funções do programa: Deletar, Editor de Minimização, Editor de Macros, Eliminação, e Simulação. Somente durante a execução de uma dessas funções, ou quando nenhuma outra função ou editor estiver ativo é possível ativar a função de ampliação e redução.

Para desativar essa função deve-se selecionar o botão com a nova imagem. Isso fará com que a função seja desativada e a imagem original volte a ser exibida. Também há a possibilidade de ativar a função de arrasto ou a oposta da função que estiver ativada para desativar automaticamente a função ativada, devido ao funcionamento dos três botões AMPLIAR, REDUZIR e ARRASTAR como botões de rádio.

4.5.9. Eliminação

A função de Eliminação retira da Área de Trabalho e do sistema criado pelo usuário portas lógicas e fios. A eliminação desses componentes é definitiva e deve-se proceder com cautela quando da aplicação dessa função.

A ativação da função é feita após selecionar o botão APAGAR, o que ativa a função e faz com que a imagem do botão seja substituída por uma que indica que a função permanece ativa.

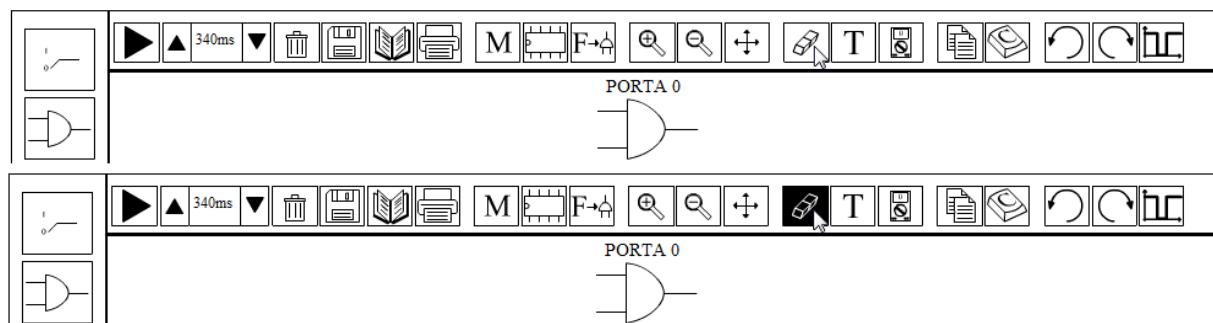


Figura 4.88 – Ativação da função de Eliminação.

Para se realizar a eliminação de portas lógicas e fios, é necessário selecioná-los, formando um grupo, que pode ter um ou mais componentes. O grupo pode ser selecionado antes ou após a ativação da função. Tendo o Grupo Selecionado deve-se posicionar o cursor do mouse sobre um de seus componentes e pressionar o botão esquerdo do mouse. Assim todos os componentes do grupo serão removidos da Área de Trabalho e do sistema.

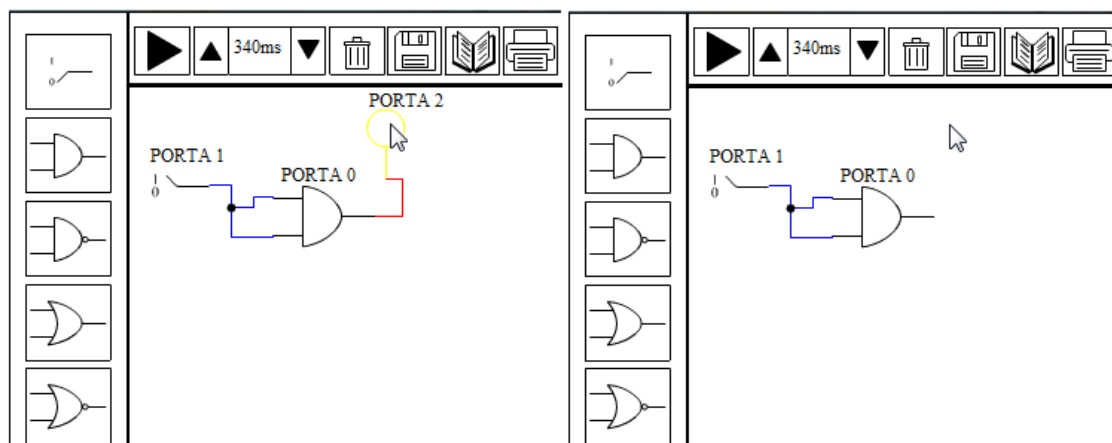


Figura 4.89 – Eliminação do Grupo Seleccionado.

As únicas funções que podem ser ativadas enquanto a função de eliminação estiver ativa são: ampliação, redução e arrasto (apenas da Área de Trabalho). Essas funções podem ser usadas para visualizar regiões da Área de Trabalho que não são exibidas na janela do programa e que possuem portas lógicas e fios que se deseja eliminar.

Para desativar a função deve-se novamente seleccionar o botão APAGAR. Assim a função é desativada e a imagem original do botão volta a ser exibida. As funções que foram ativadas enquanto a função de eliminação estava ativa, se mantêm ativas após a desativação da função de eliminação.

4.5.10. Duplicação

A função de duplicação cria uma réplica do Grupo Seleccionado criado previamente. Não há nenhum tipo de restrição quanto ao conteúdo do Grupo Seleccionado, todo o tipo de portas lógicas e fios serão replicados, inclusive suas propriedades como exibição de nome, conteúdo de memória de portas lógicas de tipo Memória e conteúdo das portas lógicas de tipo Macros serão mantidas nas réplicas dessas portas lógicas. Apenas os nomes das portas lógicas não são mantidos para haver uma diferenciação futura das novas portas lógicas criadas. No caso de memórias, o seu conteúdo será copiado para a réplica, assim como a propriedade que define o seu funcionamento como memória do tipo RAM ou ROM. No caso das portas lógicas de tipo Macros, o conteúdo da porta lógica seleccionada será copiado em sua réplica, inclusive o conteúdo e propriedades de possíveis memórias que possam fazer parte desse conteúdo, além disso, a propriedade que indica se a porta lógica de tipo Macros possui sincronia com o Editor

de Macros também é copiada. No caso de portas lógicas de tipo Pulsers, sua frequência de oscilação também é copiada para a sua réplica.

Para se realizar o processo de duplicação deve-se inicialmente selecionar o grupo que se deseja replicar e em seguida deve-se posicionar o cursor do mouse sobre o botão DUPLICAR e pressionar o botão esquerdo do mouse. Após pressionar o botão, será criada uma réplica do grupo selecionado e esta será inserida na Área de Trabalho.

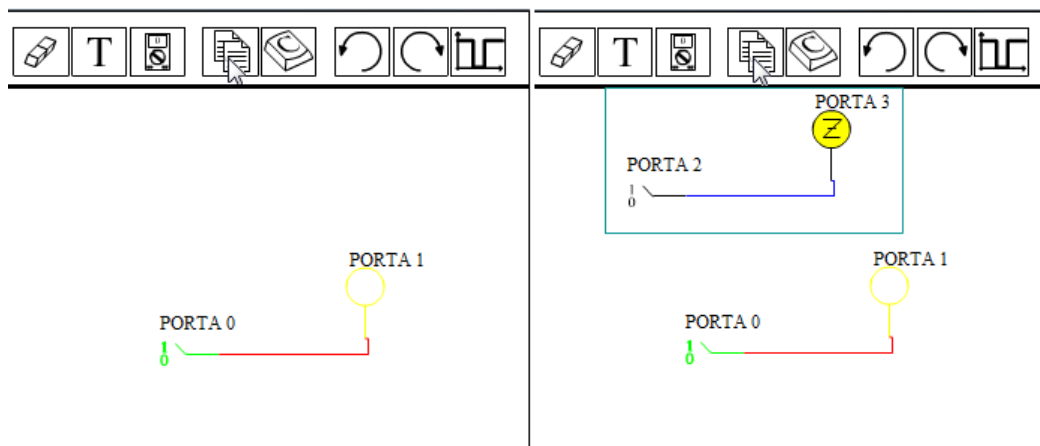


Figura 4.90 – Duplicação do grupo selecionado.

Na Figura 4.90 é possível observar que o novo grupo criado possui uma caixa ao redor de suas extremidades, para indicar qual será sua área ocupada. O novo grupo, depois de adicionado à Área de Trabalho, mantém o centro geométrico de sua área ocupada sempre junto ao cursor do mouse, assim com o movimento deste cursor o grupo é arrastado pela Área de Trabalho. Esse arrasto segue as mesmas regras de arrasto descritas no capítulo 4.5.7, ou seja, a área do grupo nunca ultrapassa os limites da Área de Trabalho, mantendo o grupo sempre no interior da Área de Trabalho. Mais uma vez o programa gerencia o arrasto e cuida para que essa regra não seja desobedecida. Neste caso a função de arrasto automático da área de trabalho encontra-se ativa.

Com o novo grupo sendo arrastado pelo cursor do mouse, deve-se posicioná-lo sobre a Área de Trabalho e pressionar o botão esquerdo do mouse, assim o processo de arrasto é desativado e a posição do grupo na Área de Trabalho será mantida como a posição no momento em que se pressionou o botão.

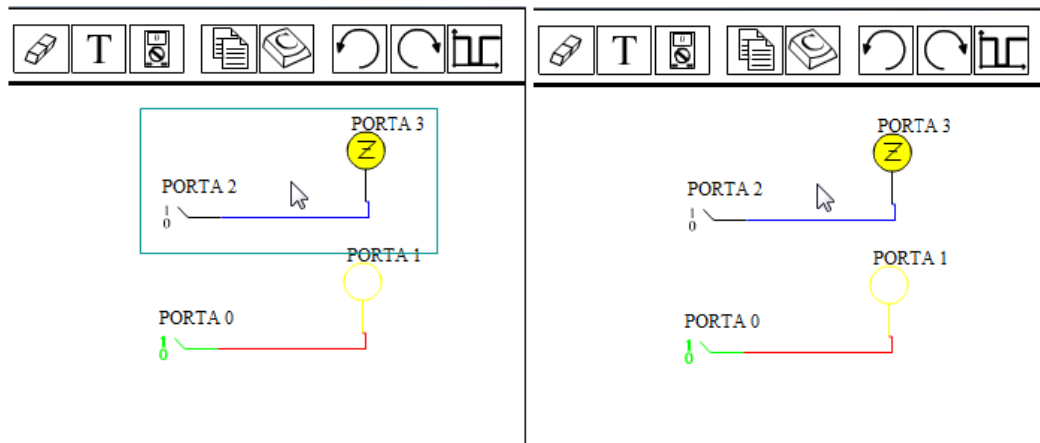


Figura 4.91 – Posicionamento na Área de Trabalho do novo grupo duplicado.

Durante o processo de arrasto do novo grupo não é possível ativar nenhuma outra função do programa e após o término do processo de arrasto, o Grupo Selecionado permanece selecionado e a caixa que fica ao redor do novo grupo criado é retirada.

Durante o processo de arrasto do grupo é possível cancelar a inserção deste grupo no sistema. O botão INICIAR/CANCELAR está exibindo nesse momento um "X" e ao selecioná-lo, o processo de arrasto será desativado e o novo grupo será eliminado.

4.5.11. Rotação

A função de rotação realiza o giro do Grupo Selecionado na direção horária ou anti-horária, sempre realizando giros de noventa graus. Não existe restrição quando ao Grupo Selecionado, podendo inclusive haver fios ligados a algum componente do grupo e a outra porta lógica ou fio que não façam parte do grupo. O programa atua gerenciando o processo de giro para manter sempre coerência entre o circuito resultante do giro e o circuito anterior ao giro. Os componentes do Grupo Selecionado se mantêm selecionado após o giro.

Para realizar o processo de rotação deve-se ter um grupo previamente selecionado, e em seguida selecionar o botão ROTAÇÃO ANTI-HORÁRIA, para giro anti-horário, ou selecionar o botão ROTAÇÃO HORÁRIA para giro horário. Após selecionar um dos botões o giro do Grupo Selecionado é executado. O pivô onde o giro será baseado é calculado como sendo a posição central da área ocupada pelos componentes do Grupo Selecionado. O giro será realizado na direção desejada e será de noventa graus. Caso as portas lógicas do Grupo Selecionado estejam com suas propriedades de exibição de nome ativa, haverá uma caixa de

texto com o nome de cada porta lógica junto a ela, assim após o giro essa caixa muda de posição para acompanhar o giro da porta lógica.

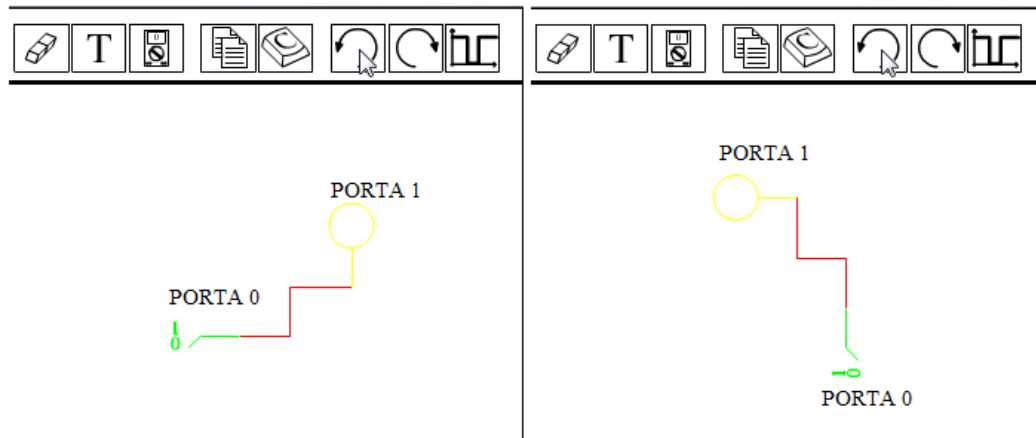


Figura 4.92 – Giro do grupo selecionado.

Após o giro do Grupo Selecionado, pode ocorrer a situação em que parte da área ocupada por este grupo esteja fora dos limites da Área de Trabalho. Como foi detalhado anteriormente essa condição nunca pode estar presente no programa, assim o programa gerencia o processo de giro e caso encontre a situação citada, atua sobre o grupo que foi girado, editando sua posição para uma em que toda a área ocupada pelos componentes do grupo estejam no interior da Área de Trabalho. Essa ação do programa ainda pode editar a posição das caixas de texto com os nomes das portas lógicas para uma nova posição em que não ultrapasse os limites da Área de Trabalho. Apenas a posição geral do grupo é editada, as posições relativas entre os componentes do Grupo Selecionado se mantêm as mesmas.

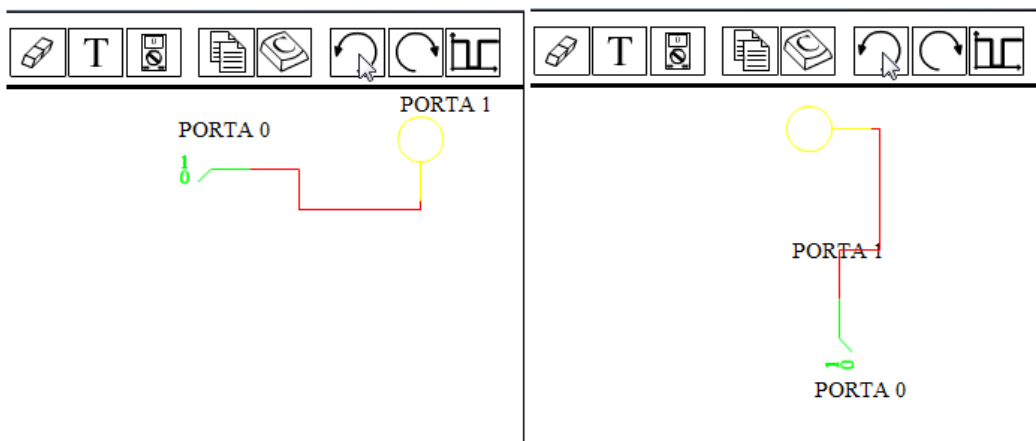


Figura 4.93 – Reposicionamento do grupo girado para não ultrapassar o limite da Área de Trabalho.

Existe um erro no projeto do programa que faz com que o reposicionamento da caixa de texto na condição da Figura 4.93, resulte em um afastamento da caixa de texto em relação à porta lógica a que essa caixa pertence.

O processo de giro de grupos resulta em um giro completo do grupo como um todo. Todos os componentes permanecem com suas propriedades inalteradas, a não ser pelo reposicionamento das caixas de texto com os nomes das portas lógicas, e em especial os fios que mantêm exatamente a mesma quantidade de quinas e arestas, e estas possuem as mesmas posições relativas. Porém no caso de fios que fazem a ligação entre algum componente do Grupo Selecionado e uma porta lógica ou fio externos ao grupo, após o giro, existe uma edição das propriedades do fio para que ele esteja em conformidade com o circuito anterior ao giro. Nesse caso novas quinas e arestas podem ser inseridas nesses fios, ou pode haver somente a edição da posição de algumas quinas e arestas desse fio. Essas edições nas propriedades do fio são feitas pelo programa segundo uma série de regras que são aplicadas a este fios para que eles continuem representando com fidelidade a ligação que existia antes da giro do grupo.

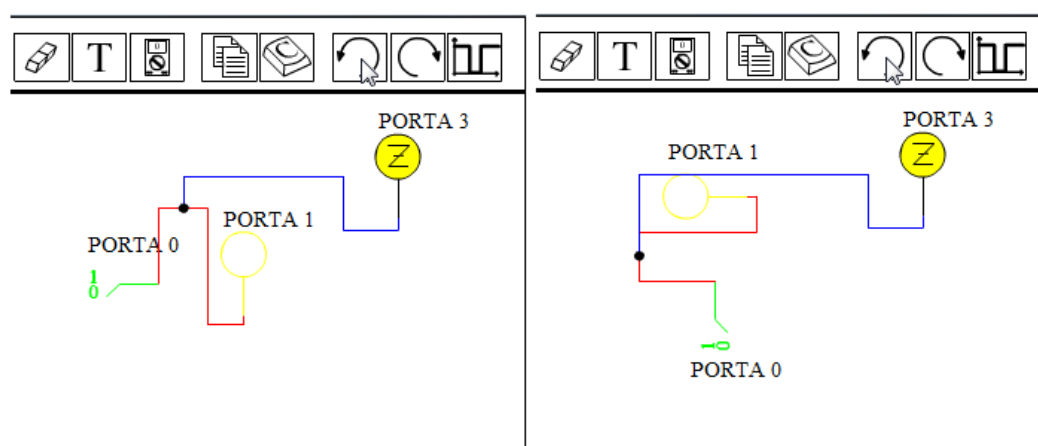


Figura 4.94 – Fio ligado ao grupo girado tem suas propriedades editadas para manter a coerência da ligação.

A edição das propriedades dos fios pode fazer com que o fio resultante seja diferente do fio original, ou até mesmo esteja disposto de maneira a encobrir o circuito girado, porém essas edições são necessárias e o programa realiza uma série de testes para garantir que o fio resultante esteja em conformidade com o circuito anterior ao giro. Cabe ao usuário do programa atuar sobre esses fios editados e posicionar suas arestas de maneira a não encobrir outros componentes do circuito.

Caso o grupo selecionado possua apenas fios, o giro não é realizado, é necessário ter pelo menos uma porta lógica no grupo selecionado.

No caso em que o grupo selecionado contenha algum fio que possua terminal em outro fio não selecionado, ou em um pino de uma porta lógica não selecionada, o giro desse grupo é realizado normalmente, todo o grupo selecionado é girado em conjunto mantendo a posição relativa dos elementos do grupo inalterada e também o fio se mantém inalterado. Porém, após o giro, o fio em questão tem suas propriedades editadas para manter a coerência da ligação que realiza com a porta lógica ou fio externo ao grupo que foi girado. Assim, após o giro, esse fio estará diferente do que era antes, e assim a área do Grupo Selecionado também será diferente. Devido a essa mudança na área do Grupo Selecionado, caso seja realizado um novo processo de giro, a posição do pivô sobre o qual esse novo giro será realizado, será diferente da posição do pivô sobre o qual o último giro foi realizado. Essa diferença entre a posição dos pivôs provoca uma situação de deslocamento do Grupo Selecionado, e essa situação se repetirá caso novos processos de giro sejam executados, podendo fazer com que o grupo selecionado se afaste da sua posição original.

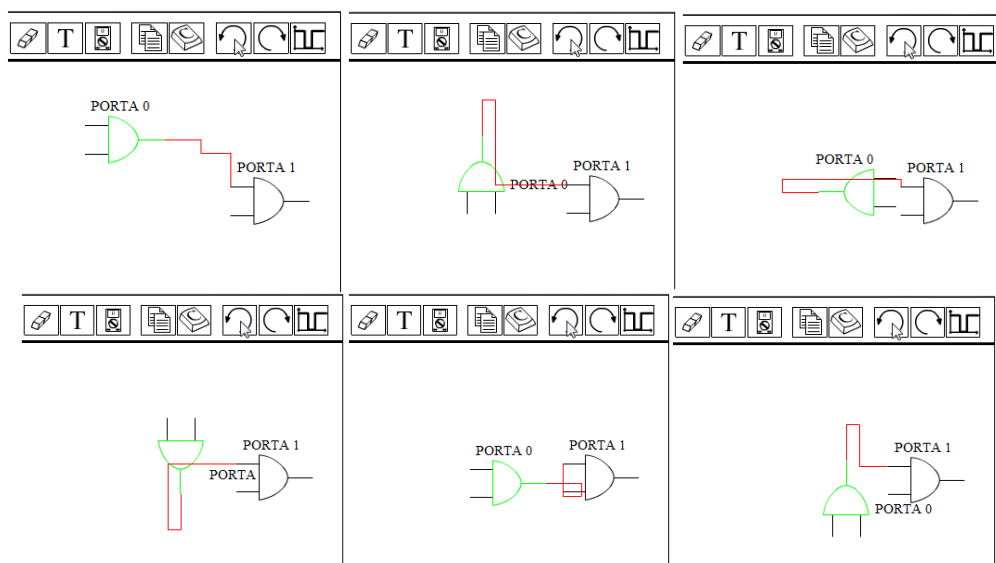


Figura 4.95 – Após sucessivos processos de giro o Grupo Selecionado se afasta de sua posição original.

No caso da Figura 4.95 é possível que a situação de deslocamento do Grupo Selecionado convirja para uma posição em que os pivôs dos processos de giro sejam os mesmos e assim não haja mais deslocamento.

Em um Grupo Selecionado que contenha fios, esses fios só participam do processo de giro se pelo menos uma das portas lógicas ou fios com os quais esse fio faz ligação estiverem selecionados, caso contrário, o fio não participará do giro e apenas poderá ser redimensionado caso seja necessário.

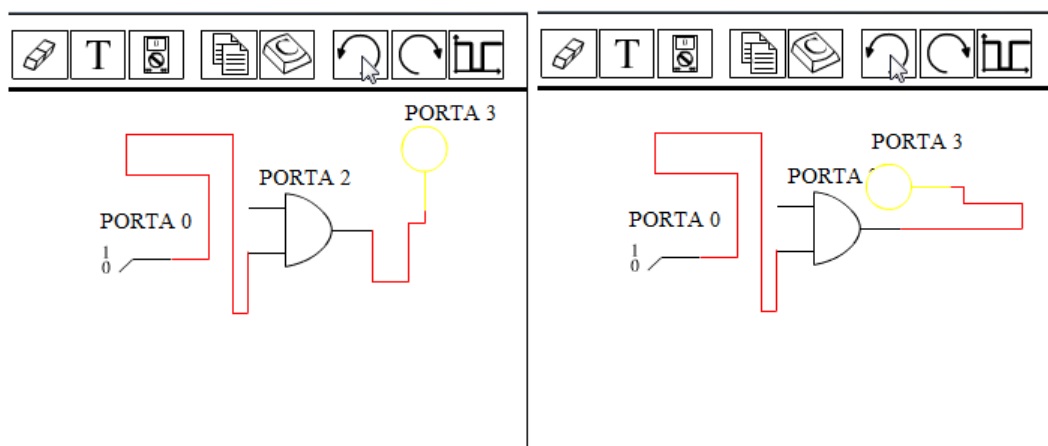


Figura 4.96 – Fio que não tem alguma das portas lógicas ou fios de seus terminais selecionados não participa do giro.

Assim como no caso do arrasto de fios que possuem outros fios ligados a eles, após a edição das propriedades do fio, o programa se encarrega de manter a informação visual da ligação entre os fios coerente com o circuito anterior ao giro.

4.5.12. Deletar

A função Deletar realiza a reinicialização do programa, ela faz com que o programa retorne ao estado inicial de quando foi executado pelo usuário, todas as informações de portas lógicas e fios são apagadas, os dados dos editores também são apagados, apenas informações sobre atalhos criados são mantidas.

Para se iniciar a função Deletar, deve-se selecionar o botão DELETAR e nesse momento a imagem do botão é substituída por uma que indica que a função está ativa. Com a função iniciada, deve-se mais uma vez selecionar o botão DELETAR, agora com imagem diferente, para se confirmar a execução da função Deletar. Nesse momento todas as portas lógicas e fios presentes na Área de Trabalho são retirados e o botão DELETAR volta a exibir a imagem original.

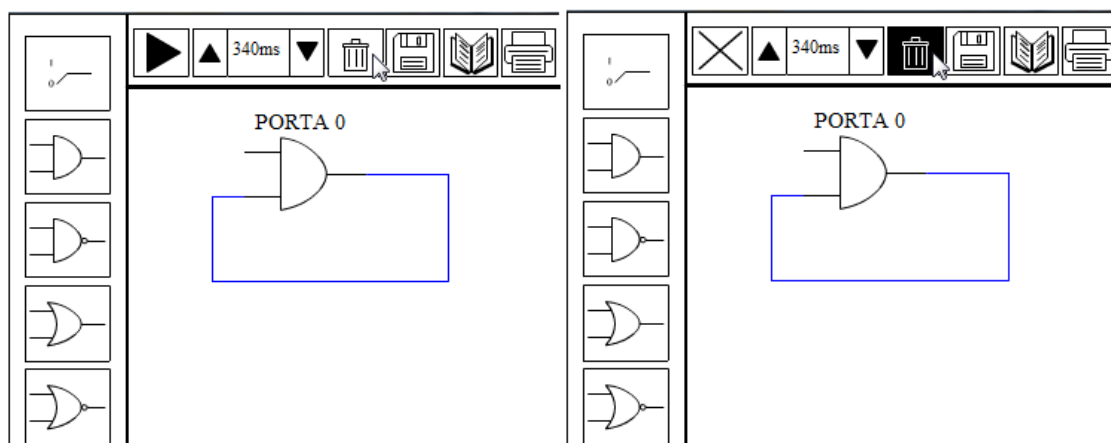


Figura 4.97 – Deve-se confirmar a execução da função selecionando o botão DELETAR novamente após iniciá-la.

A execução da função Deletar provoca a remoção definitiva de todas as informações criadas pelo usuário do programa, não sendo possível recuperar essas informações por nenhum meio.

Depois da inicialização da função não é possível ativar nenhuma outra função no programa, uma vez que o programa espera a confirmação da execução da função Deletar, apenas as funções Ampliar, Reduzir e Arrasto, são permitidas, pois podem ser utilizadas para visualizar todas as regiões da Área de Trabalho, em uma possível análise do circuito para checar se a função Deletar pode mesmo ser executada.

É possível cancelar a função Deletar após sua inicialização. Depois da inicialização, o botão INICIAR/CANCELAR apresenta a imagem de um "X", e basta selecioná-lo para finalizar a função.

4.5.13. Salvar

A função Salvar é responsável por salvar todos os dados do sistema criado em um arquivo de dados. Todas as informações do circuito são salvas, como posição, nome e orientação das portas lógicas, informações do Editor de Propriedades das portas lógicas, estado das chaves lógicas, posição e formato dos fios, todos os circuitos criados no Editor de Macros e todas as informações criadas no Editor de Atalhos.

Para se iniciar a função Salvar deve-se selecionar o botão SALVAR, e nesse momento o botão SALVAR muda sua imagem, a janela de salvamento de arquivos do sistema operacional é chamada e o programa fica inacessível enquanto essa janela não for encerrada. Ao encerrar a janela de salvamento de arquivos, o programa volta a ser acessível e o botão SALVAR retorna a exibir a sua imagem original.

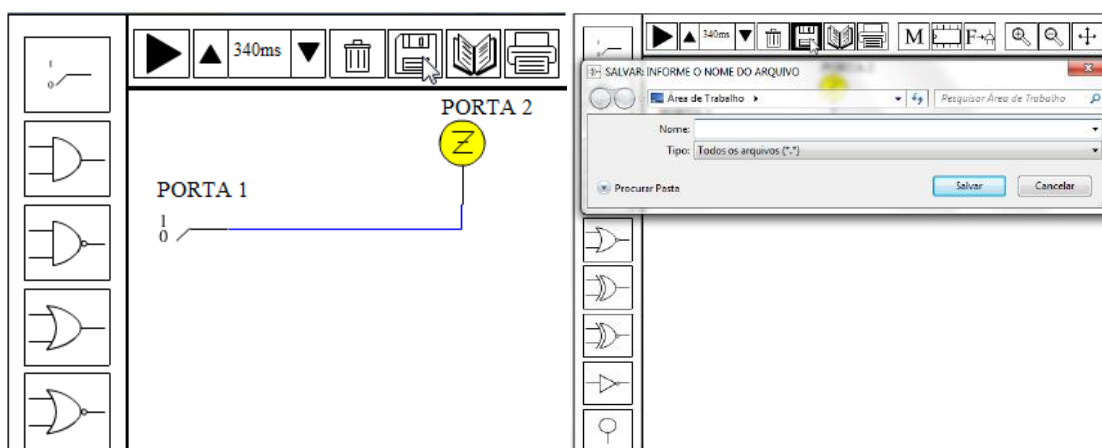


Figura 4.98 – Abertura da janela de salvamento de arquivos do sistema operacional ao iniciar a função Salvar.

Caso o salvamento seja confirmado será criado um arquivo de dados com o nome que foi indicado na janela de salvamento de arquivos, e com a extensão ".ccm". Esse arquivo de dados pode ser aberto em um programa de editor de textos, porém não é recomendado realizar edições nesse arquivo, pois o arquivo pode ser corrompido caso alguma informação seja editada de maneira errada. Não há uma otimização na exibição dos dados do sistema nesse arquivo de dados, assim a compreensão do sistema pela análise do arquivo de dados em um programa de edição de textos é impossível sem ter a informação de como os dados estão dispostos no arquivo. Essa informação é restrita ao código do programa deste projeto e assim somente através dele é possível interpretar de maneira correta o arquivo de dados.

O arquivo de dados pode ser renomeado através da utilização de algum programa do sistema operacional, não havendo qualquer tipo de limitação para o nome do arquivo, apenas é necessário manter a extensão ".ccm", pois sem ela é necessário fornecer exatamente o nome do arquivo para que este possa ser carregado pelo programa.

Existe no programa um gerenciador que, após o carregamento de um arquivo de dados, checka quando alguma modificação no sistema é feita, deixando-o diferente do arquivo carregado. Assim, no caso do fechamento da janela do programa, acidental ou

propositadamente, se esse gerenciador indicar que o sistema presente no programa difere daquele carregado anteriormente, o programa chama a janela de salvamento de arquivos do sistema operacional para que o sistema seja salvo. Não é obrigatório realizar esse salvamento e o encerramento da janela de salvamento de arquivos encerra o programa.

4.5.14. Abrir

A função Abrir é responsável por realizar o carregamento de um arquivo de dados criado anteriormente pelo programa, e da criação dos circuitos na Área de Trabalho, dos circuitos do Editor de Macros e dos atalhos do Editor de Atalhos.

Para se iniciar a função Abrir deve-se selecionar o botão ABRIR, e nesse momento o botão ABRIR muda sua imagem, a janela de carregamento de arquivos do sistema operacional é chamada e o programa fica inacessível enquanto essa janela não for encerrada. Ao encerrar a janela de salvamento de arquivos, o programa volta a ser acessível e o botão ABRIR retorna a exibir a sua imagem original.

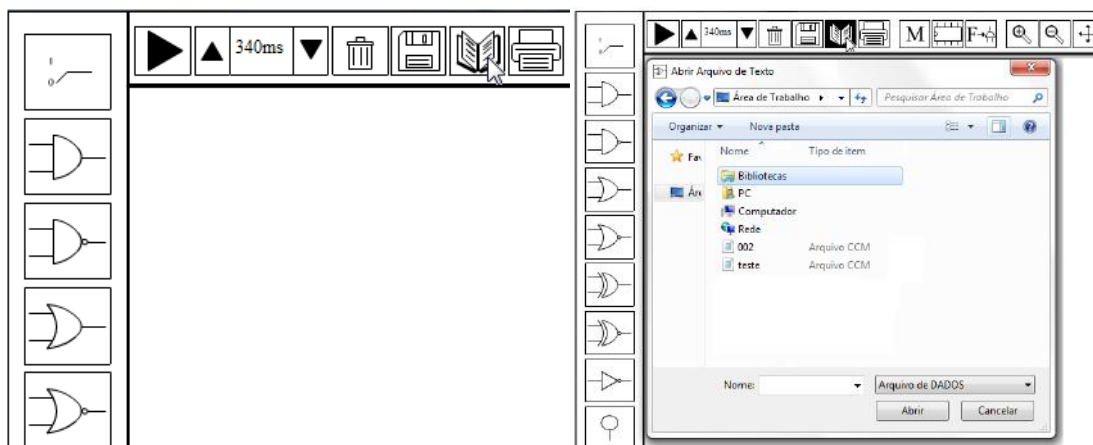


Figura 4.99 – Abertura da janela de carregamento de arquivos do sistema operacional ao iniciar a função Abrir.

Quando a janela de carregamento de arquivos do sistema operacional é chamada, é informado a ela que somente arquivos com a extensão ".ccm" devem ser exibidos. Isso é observado na janela, porém é possível informar o nome do arquivo que se quer carregar através do teclado, mesmo que este arquivo não possua a extensão ".ccm". Quando o carregamento é confirmado, uma série de checagem é aplicada sobre o arquivo indicado pela janela de carregamento, e a primeira delas é a checagem da extensão do arquivo, onde

somente arquivos com a extensão ".ccm" podem ser carregados. Caso o arquivo possua outra extensão uma mensagem de erro será exibida informando que o arquivo não pode ser carregado.

Caso o arquivo carregado possua a extensão ".ccm", iniciam-se uma série de testes para checar a consistência dos dados contidos nesse arquivo. Esses testes visam encontrar possíveis edições no arquivo de dados realizadas em outros programas, que provocam um quadro de perda de informações ou até de inconsistência dos dados carregados. Caso esse tipo de edição seja confirmada pelo programa, o processo de carregamento é interrompido e uma mensagem de erro é exibida informando que o arquivo não pode ser carregado pois encontra-se corrompido.

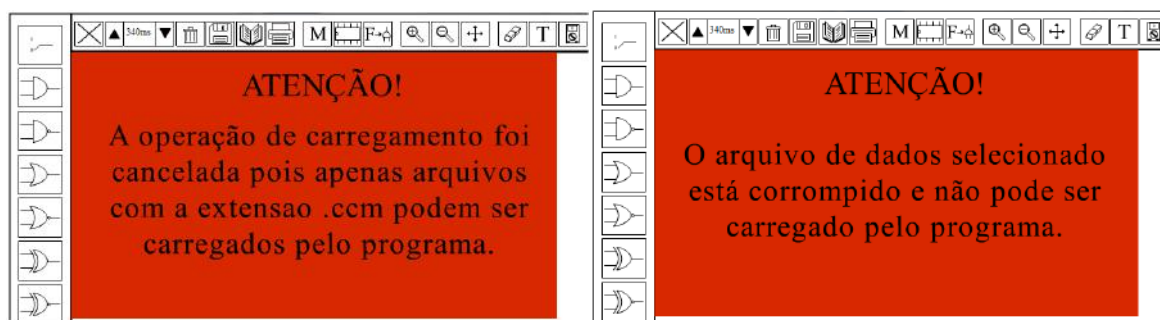


Figura 4.100 – Aviso de arquivo com extensão não suportada e arquivo corrompido.

Caso não haja nenhum tipo de edição no arquivo carregado, o programa recria todos os circuitos e carrega nos editores todas as informações do arquivo de dados. O processo de carregamento se encerra e o programa fica pronto para ser utilizado.

O gerenciador que procura pela existência de edições que possam corromper o arquivo de dados é um processo robusto que consegue filtrar quase que a totalidade dos arquivos corrompidos, porém existe uma situação que esse gerenciador ainda não está preparado para lidar. Essa situação é muito específica e requer que a edição no arquivo de dados seja realizada de maneira a burlar o filtro do gerenciador. É necessário que o autor das edições conheça bem a estrutura do arquivo de dados para que essa edição seja bem sucedida, caso contrário o autor deve contar com a sorte para que a edição seja bem sucedida. Caso essa edição seja bem sucedida em burlar o gerenciador, durante a execução do programa podem ocorrer situações de travamentos ou situações de desaparecimento de portas lógicas ou fios, ou de mau funcionamento do Editor de Macros. Mais uma vez é recomendado que o conteúdo dos arquivos de dados criado pelo programa não sejam editados em outros programas, para evitar os problemas citados.

4.6. Editores

4.6.1. Editor de Propriedades

Através deste editor é possível editar as propriedades das portas lógicas adicionadas na Área de Trabalho. Esse editor também tem um papel informativo, pois apresenta informações sobre cada porta lógica existente no programa, exibindo uma imagem ilustrativa que contém informações como nome da porta lógica, a função que aquela porta lógica representa e a tabela verdade dessa função. Em algumas portas lógicas são exibidas imagens que informam a equivalência da porta lógica com outro circuito.

Para se iniciar o editor deve-se posicionar o cursor do mouse sobre uma porta lógica e pressionar o botão esquerdo do mouse. É necessário que não haja portas lógicas e fios selecionados quando essa seleção é realizada, e que nenhuma outra função ou editor estejam ativos, ou que somente o Editor de Macros esteja ativo. Quando o editor é iniciado, sua janela é inserida na Área de Trabalho encobrendo os circuitos que lá estejam. A janela encontra-se posicionada no canto superior esquerdo da janela do programa, abaixo da Barra de Ferramentas e a direita do Container de Portas, para qualquer tamanho e posicionamento da Área de Trabalho.

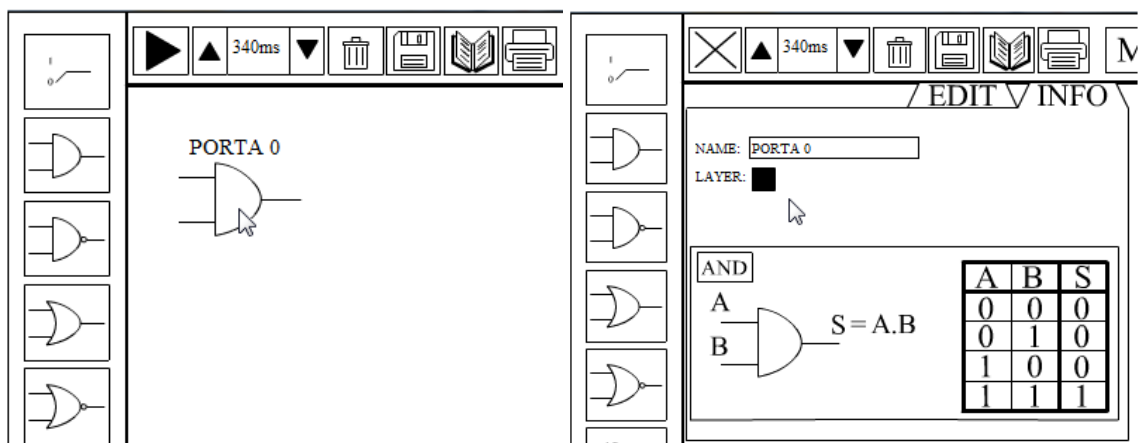


Figura 4.101 – Inicialização do Editor de Propriedades. A janela do editor é inserida na Área de Trabalho.

Não é possível ativar nenhuma função enquanto o editor estiver ativo, é possível apenas fechá-lo selecionando o botão INICIAR/CANCELAR que exibe a imagem de um "X" após a

inicialização do editor. Caso seja fechado, a janela do editor é removida da Área de Trabalho e o botão INICIAR/CANCELAR volta a exibir sua imagem original.

A janela do editor é composta por duas abas, a aba "INFO" e a aba "EDIT". A aba "INFO" é sempre exibida quando se inicia o editor. Na aba "INFO" é possível visualizar e editar o nome da porta lógica que é exibido na caixa de texto. Essa informação encontra-se na janela do editor a direita do texto "NAME", dentro de um retângulo que delimita a região onde o nome da porta lógica deve ser inserido, e para modificá-lo deve-se posicionar o cursor do mouse sobre esse retângulo e pressionar o botão esquerdo do mouse. Nesse momento é possível modificar o nome exibido através do teclado. Existe um limite máximo de vinte caracteres para o nome da porta lógica e apenas o caractere "*" não pode ser utilizado. É permitido manter o nome com nenhum caractere.

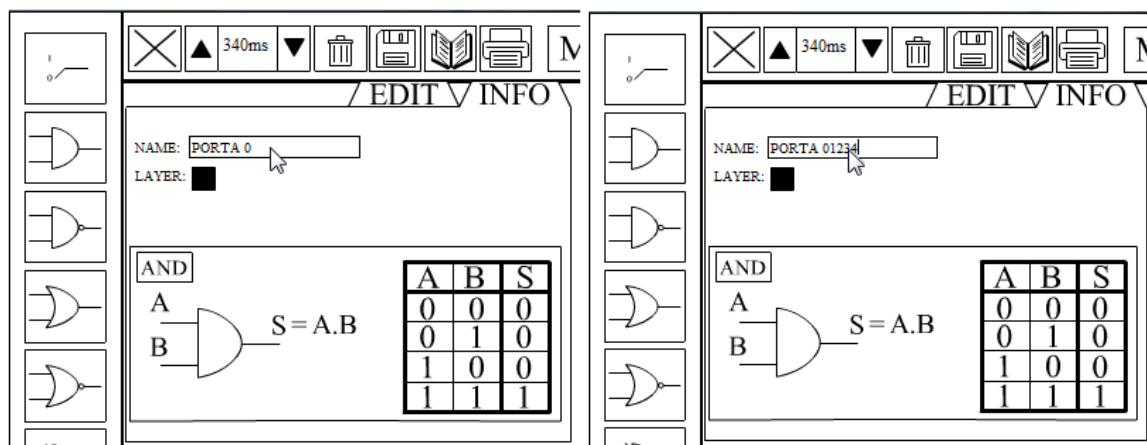


Figura 4.102 – Edição do nome da caixa de texto da porta lógica.

Ainda na aba "INFO" existe um marcador que está ligado à opção de exibição da caixa de texto com o nome da porta lógica. Esse marcador encontra-se à direita do texto "LAYER", e é representado por um quadrado preenchido de preto. Ele inicia ativado, o que faz com que a caixa de texto seja exibida próxima à porta lógica. Para desativar a opção deve-se posicionar o cursor do mouse sobre a imagem do marcador e pressionar e soltar o botão esquerdo do mouse, assim a imagem do marcado muda para a de um quadrado preenchido de branco, e a caixa de texto deixa de ser exibida. Para reativar a opção deve-se repetir o processo acima, e assim a imagem original volta a ser exibida.

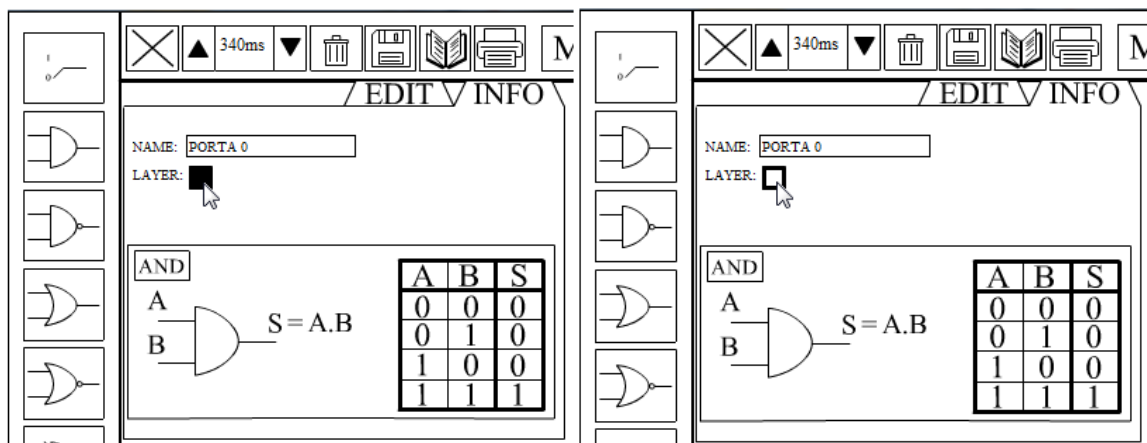


Figura 4.103 – Edição do marcador de exibição da caixa de texto da porta lógica.

Para as portas lógicas de tipo Memória, ainda existe um segundo marcador que indica o funcionamento da memória como RAM ou ROM. Esse marcador encontra-se à direita do texto "RAM", e é representado por um quadrado preenchido de preto quando está ativado, ou por um quadrado preenchido de branco quando não está ativo. As portas lógicas de tipo Memória iniciam com essa opção ativada se for do tipo RAM, ou desativada se for do tipo ROM. O funcionamento desse marcador é o mesmo do marcador de exibição da caixa de texto. Para se ativar ou desativar essa opção, deve-se posicionar o cursor do mouse sobre o marcador e pressionar e soltar o botão esquerdo do mouse.

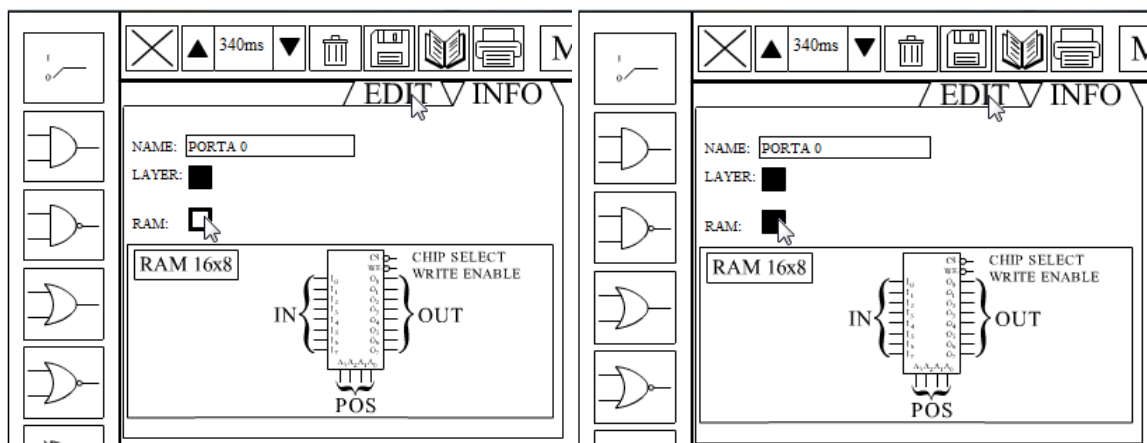


Figura 4.104 – Edição do marcador de opção RAM da caixa de texto da porta lógica de tipo Memória.

Para as portas lógicas de tipo Macros, o segundo marcador está ligado à opção de sincronização entre a porta lógica e o Editor de Macros. Esse marcador encontra-se à direita do texto "SYNC", ele sempre inicia desativado, e é representado pela imagem de um quadrado preenchido de branco. Para ativar essa opção deve-se posicionar o cursor do mouse sobre o marcador e pressionar e soltar o botão esquerdo do mouse, assim a imagem do

marcador muda para a de um quadrado preenchido de preto. Para desativar o marcador, o mesmo processo deve ser seguido.

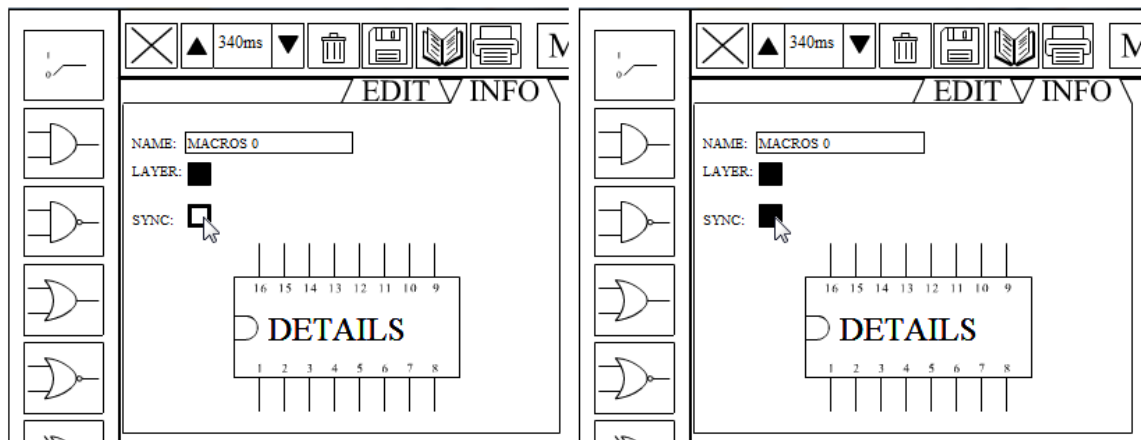


Figura 4.105 – Edição do marcador de opção SYNC da caixa de texto da porta lógica de tipo Macros.

A imagem ilustrativa da porta lógica encontra-se na parte inferior da aba "INFO" e apresenta informações das portas lógicas como nome, função, tabela da função, equivalência da porta lógica com outros circuitos, ou imagens ilustrativas da porta lógica. Não há nenhuma forma de interação com essas imagens. Nas Figuras 4.103 e 4.104 é possível visualizar essas imagens e as informações que elas apresentam.

No caso das portas lógicas de tipo Macros, a imagem ilustrativa não está presente, e em seu lugar existe uma imagem de um Circuito Integrado com o nome "DETAILS", assim como pode ser observado na Figura 4.105. Ao posicionar o cursor do mouse sobre essa imagem e pressionar o botão esquerdo do mouse, a Janela de Visualização dos circuitos internos da porta lógica é inserida na Área de Trabalho. Essa janela contém os circuitos internos da porta lógica.

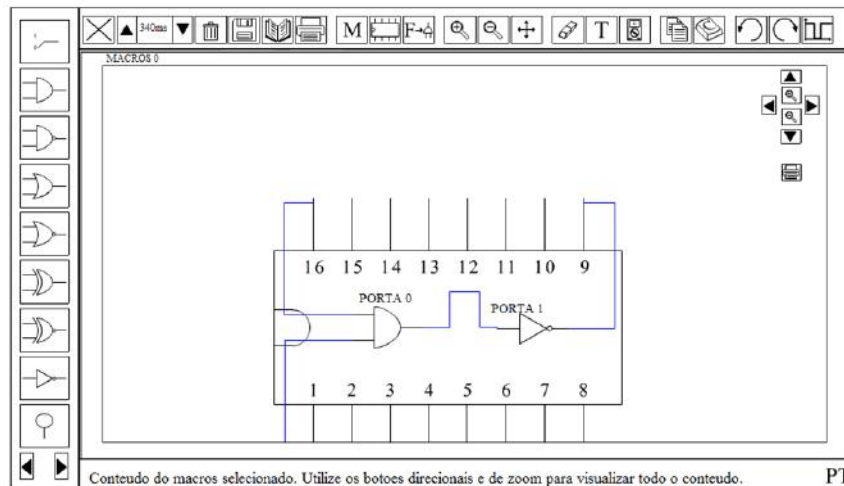


Figura 4.106 – Janela de Visualização dos circuitos internos da porta lógica de tipo Macros.

Na janela ilustrada na Figura 4.106 está presente uma segunda área de trabalho contendo uma representação de um circuito integrado de dezesseis pinos e os circuitos internos da porta lógica sobre a qual o editor foi iniciado. Essa área de trabalho da janela possui as mesmas dimensões da Área de Trabalho do programa, 6000 pixels por 6000 pixels, e existe nessa janela um conjunto de botões que permite a navegação sobre essa área, e também a ampliação e redução da área de trabalho da janela. Os botões com imagens de setas apontando para esquerda, direita, cima e baixo são responsáveis por realizar o arrasto da área de trabalho da janela no sentido oposto ao que suas imagens estão direcionadas. Basta posicionar o cursor do mouse sobre um desses botões, pressionar e manter pressionado o botão esquerdo do mouse para que o arrasto seja realizado até que o cursor do mouse seja movido para fora do botão ou que o botão esquerdo do mouse seja liberado. A Figura 4.107 ilustra o processo de arrasto da área de trabalho da janela apresentada na Figura 4.106.

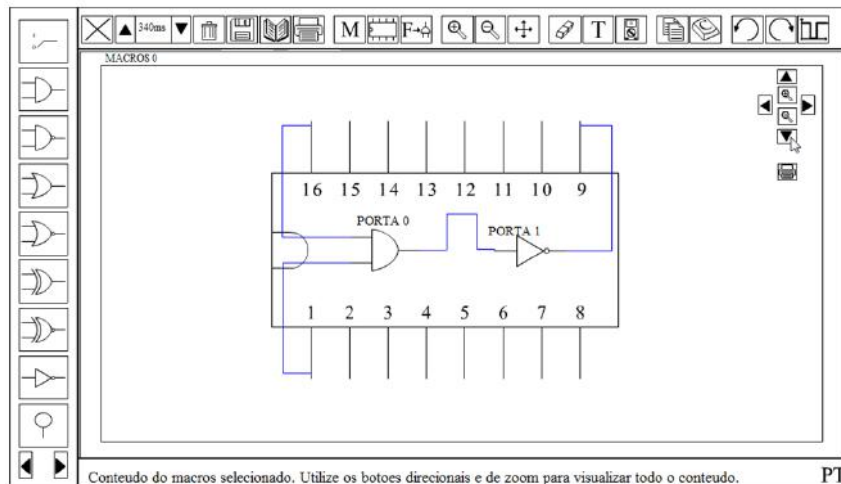


Figura 4.107 – Seleção do botão com seta para baixo move a área de trabalho da janela para cima.

Os botões com imagem de lupa contendo um sinal de mais e sinal de menos, são os botões responsável por realizar a ampliação e redução da área de trabalho da janela, respectivamente. Para realizar a ampliação e redução basta posicionar o cursor do mouse sobre um desses botões e pressionar o botão esquerdo do mouse. Esse processo pode ser realizado reduzindo a área de trabalho da janela a um tamanho mínimo de 500 pixels por 500 pixels, enquanto o tamanho máximo continua sendo 6000 pixels por 6000 pixels. Esse novo limite mínimo permite visualizar uma porção maior da área de trabalho da janela. A Figura 4.108 ilustra o processo de redução das dimensões da área de trabalho da janela apresentada na Figura 4.106.

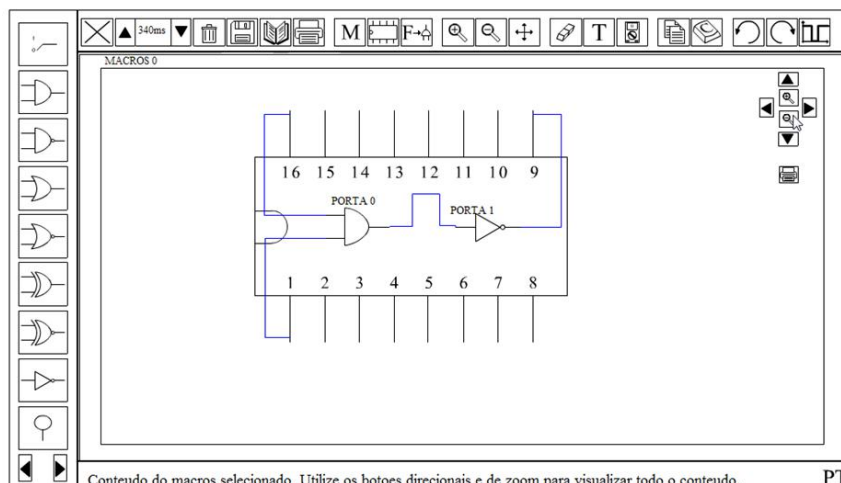


Figura 4.108 – Seleção do botão REDUZIR diminui o tamanho da área de trabalho da janela de visualização.

Existe ainda um botão responsável por realizar a impressão do circuito interno da porta lógica, esse botão possui a imagem de uma impressora, assim como o botão IMPRIMIR, e

para iniciar o processo de impressão deve-se selecionar esse botão. O processo descrito inicia o Editor de Impressão que interage com a área de trabalho da janela. O funcionamento desse editor é detalhado no capítulo 4.6.6.

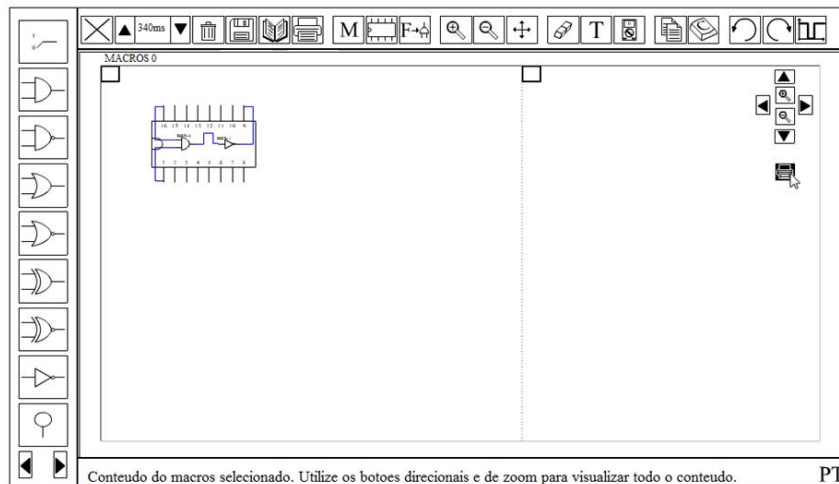


Figura 4.109 – Seleção do botão de impressão inicializa o processo de impressão do circuito exibido na janela.

Para fechar a Janela de Visualização dos circuitos internos da porta lógica, deve-se posicionar o cursor do mouse sobre a área de trabalho da janela e pressionar o botão esquerdo do mouse, assim essa janela é retirada e a janela do editor volta a ser apresentada. Nas proximidades dos botões da janela descritos acima existe uma proteção para que não se feche a janela por acidente.

Para exibir a aba "EDIT" deve-se posicionar o cursor do mouse sobre a mesma e pressionar e soltar o botão esquerdo do mouse. A aba "EDIT" passa a ser exibida sobre a aba "INFO". É possível selecionar qualquer uma das abas a qualquer momento, e as informações contidas nestas abas se mantêm as mesmas, mesmo se elas forem modificadas e não confirmadas.

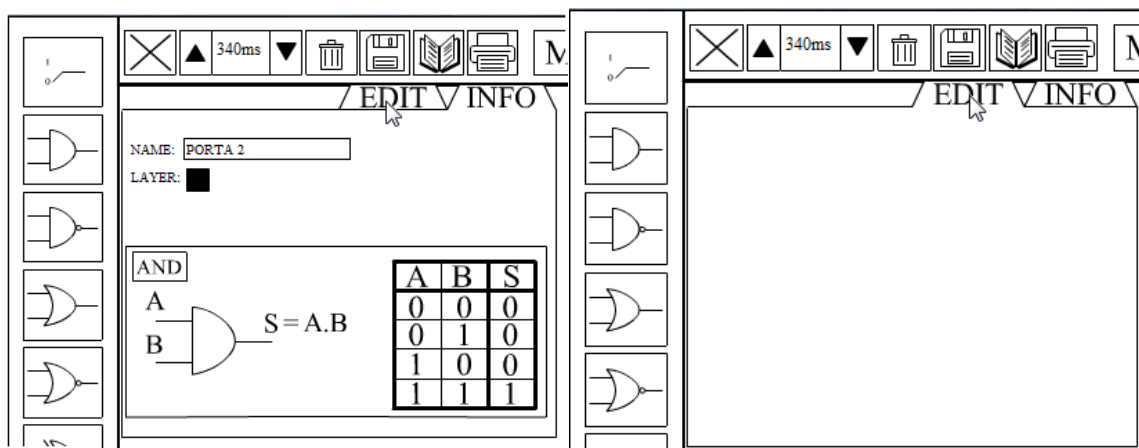


Figura 4.110 – Seleção da aba EDIT na janela do editor.

A aba "EDIT" só contém informações para as portas lógicas de tipo Memória e Pulser. Para os outros tipos de porta lógica essa aba encontra-se vazia, assim como é ilustrado na Figura 4.110. Nas portas lógicas de tipo Pulser, essa aba contém um conjunto de botões mais mostrador, onde o mostrador exibe qual é o período de atualização do estado lógico de suas saídas, e as setas próximas ao mostrador são responsáveis por editar o valor desse período. O período de atualização da porta lógica é baseado no período de atualização da execução da simulação, sendo sempre calculado como o resultado da multiplicação de uma variável por esse período. O valor inicial da variável é três, ou seja, o período de atualização da porta lógica é igual a três vezes o valor do período de atualização da execução da simulação. Os botões com imagem de seta apontando para baixo e para cima, quando selecionados, são responsáveis por retirar e acrescentar, respectivamente, uma unidade a variável que multiplica o período de atualização da execução da simulação no cálculo do período de atualização da porta lógica.

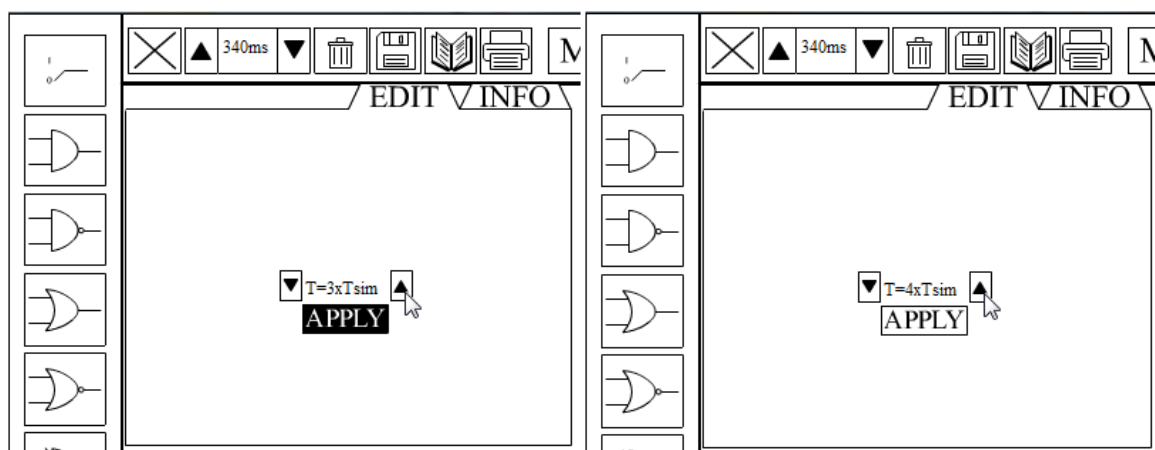


Figura 4.111 – Edição no período de atualização, na aba EDIT de uma porta lógica de tipo Pulser.

O botão APPLY é responsável por confirmar as modificações realizadas no período de atualização da porta lógica. Esse botão inicia com uma imagem com fundo de cor preta, e assim que alguma modificação é feita no valor do período de atualização da porta lógica, esse botão tem sua imagem modificada para uma que apresenta fundo de cor branca, indicando que é necessário selecionar esse botão para que as modificações possam ser validadas, como pode ser observado na Figura 4.111. O botão permanece com a imagem alterada mesmo se outra aba for selecionada. Caso o editor seja encerrado enquanto o botão ainda estiver exibindo a imagem alterada, as modificações no período de atualização da porta lógica não são aplicadas, e este mantém o valor que tinha antes da inicialização do editor. Para validar as modificações deve-se posicionar o cursor do mouse sobre o botão APPLY, que exibe uma imagem de fundo branco, e pressionar o botão esquerdo do mouse. Após validar as modificações, o botão volta a exibir a imagem original.

Nas portas lógicas de tipo Memória, a aba "EDIT" apresenta uma tabela contendo todas as posições da memória e todos os dígitos dessas posições, um conjunto de botões para navegar pelas posições dessa tabela, um conjunto de botões para impor valores à tabela, um botão de impressão e um botão APPLY para validação das modificações na tabela.

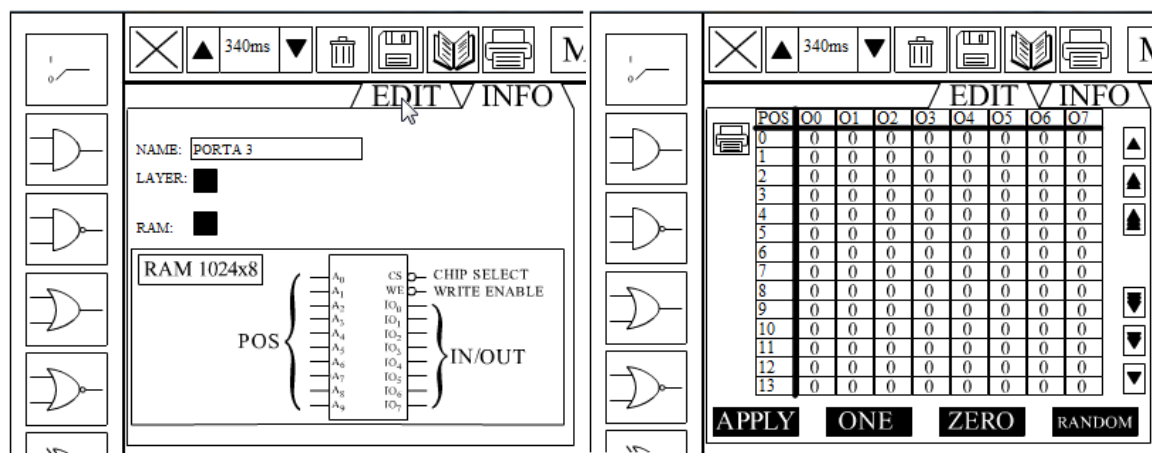


Figura 4.112 – Seleção da aba EDIT de uma porta lógica de tipo Memória.

A tabela apresentada possuirá um valor máximo de quatorze linhas representando posições da memória mais uma linha de cabeçalho, e o número de colunas será igual ao número de dígitos que cada posição da memória possuir mais uma coluna de identificação do número da posição. As células de cada linha são preenchidas com as informações dos dígitos de cada posição da memória, respeitando sempre a ordem em que esses dígitos são dispostos segundo a primeira linha de cabeçalho, ou seja, o dígito menos significativo é exibido na coluna "O0" e o dígito mais significativo é exibido na coluna "O7". Para se modificar o valor

exibido em uma célula deve-se posicionar o cursor do mouse sobre essa célula e pressionar o botão esquerdo do mouse, mudando de valor lógico zero para um, ou de um para zero.

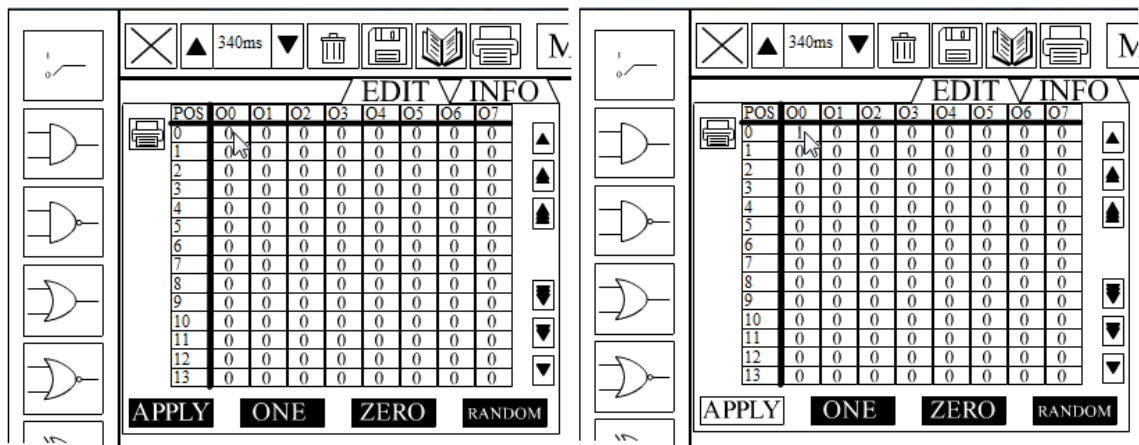


Figura 4.113 – Edição do valor de uma célula da tabela.

Pode existir caso em que a quantidade de posições da memória seja muito grande e não possa ser exibida dentro da janela do editor, que suporta uma quantidade máxima de quatorze linhas. Assim é preciso recorrer aos botões de navegação que se encontram à direita da tabela. A tabela sempre inicia exibindo na segunda linha a primeira posição da memória representada na coluna "POS" pelo número zero, e nas linhas seguintes mostra as demais posições da memória até a décima quinta linha que exibe a décima quarta posição da memória, que é representada na coluna "POS" pelo número treze. Os botões de navegação são utilizados para fazer com que outras posições da memória sejam exibidas nas linhas da tabela. Os botões com imagens de setas direcionadas para cima fazem com que a tabela seja atualizada para mostrar posições da memória de menor número de identificação e os botões com imagens de setas direcionadas para baixo, posições da memória de maior número de identificação.

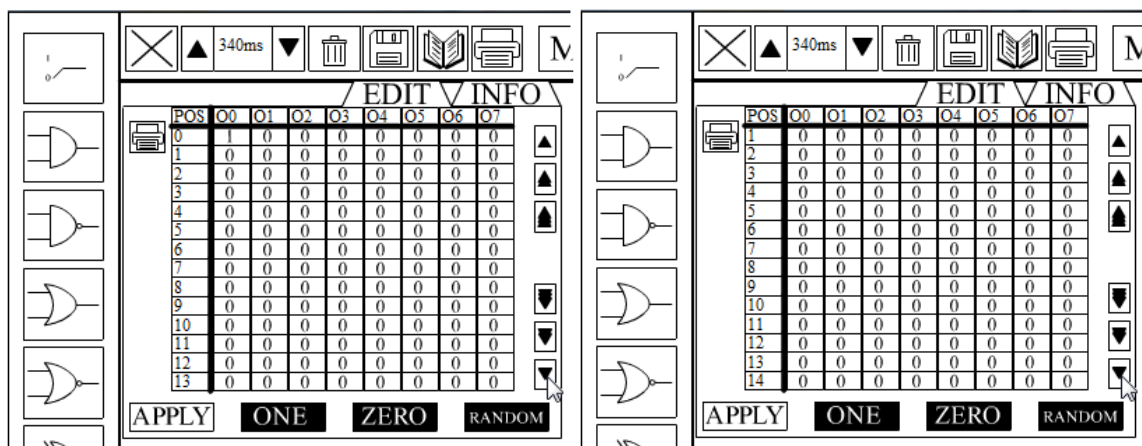


Figura 4.114 – Atualização das posições da memória exibidas na tabela.

Na Figura 4.114 o botão que apresenta imagem com apenas uma seta direcionada para baixo é selecionado. Esse botão faz com que a tabela seja atualizada para exibir em suas linhas posições da memória iniciando da posição de número de identificação um e terminando na posição da memória de número de identificação quatorze, ou seja, houve a adição de uma unidade ao número de identificação da posição da memória que ocupa cada uma das linhas da tabela em relação ao número de identificação da posição da memória que anteriormente ocupava a mesma linha. O mesmo processo ocorre caso os botões com duas ou três setas direcionadas para baixo fossem selecionados, porém haveria a adição de respectivamente dez e cem unidades ao número de identificação da posição da memória que ocupa cada linha da tabela. No caso da Figura 4.114, as posições exibidas iniciariam pela posição de número de identificação igual a dez e encerrariam pela posição de número de identificação igual a vinte e três, para o botão com duas setas e para o botão com três setas, o número de identificação da primeira posição exibida seria cem e da última posição exibida seria cento e treze. Assim é possível exibir posições de memória que possuam número de identificação elevado sem a necessidade de ter que percorrer todas as posições entre as iniciais e elas.

A seleção dos botões com setas direcionadas para cima resulta em um processo semelhante ao descrito acima, porém com a diferença que há a retirada de uma, dez ou cem unidades do número de identificação da posição da memória que ocupa cada uma das linhas da tabela em relação ao número de identificação da posição da memória que anteriormente ocupava a mesma linha.

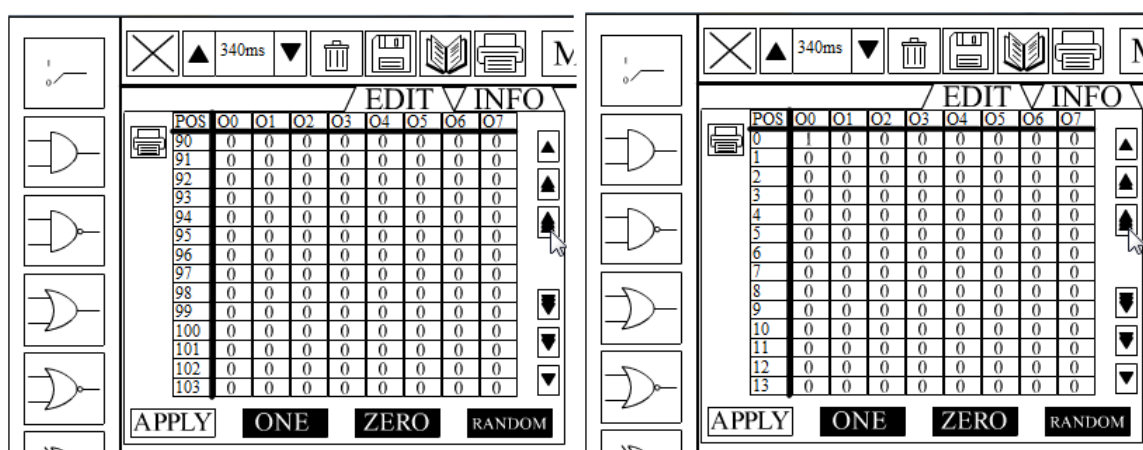


Figura 4.115 – Atualização das posições da memória exibidas na tabela.

No caso da Figura 4.115 o processo de seleção do botão com três setas direcionadas para cima resultaria na retirada de cem unidades do número de identificação de cada uma das posições exibidas na tabela. Assim para a primeira posição da memória exibida, que tem

número de identificação igual a noventa, a retirada de cem unidades resultaria na exibição da posição da memória de número de identificação igual a menos dez, e essa posição não existe. Como forma de prevenir esse tipo de situação, a retirada de cem unidades é cancelada e é exibida a primeira posição da memória, de número de identificação zero, no lugar na primeira posição exibida anteriormente. Assim toda vez que a seleção de algum botão ocasionar a exibição de posições da memória que não existem, serão exibidas as quatorze primeiras posições da memória, para o caso de botões com setas apontadas para cima, e as quatorze últimas posições da memória, para o caso de botões com setas apontadas para baixo.

Se a memória possuir menos de quinze posições, os botões de navegação são bloqueados.

Os botões ONE e ZERO, quando selecionados, inserem em todas as células da tabela os valores um e zero, respectivamente. O botão RANDOM insere valores aleatórios a cada uma das células da tabela. O botão APPLY, assim como descrito no caso das portas lógicas de tipo Pulser, inicia com uma imagem de fundo de cor preta, e tem essa imagem substituída por uma de cor branca quando alguma modificação é feita em alguma célula da tabela, como pode ser observado na Figura 4.113, e permanece assim mesmo que outra aba seja selecionada, assim como as modificações na tabela também são mantidas. Ao selecionar o botão APPLY todas as mudanças na tabela são validadas e o botão volta a exibir a imagem com fundo de cor preta.

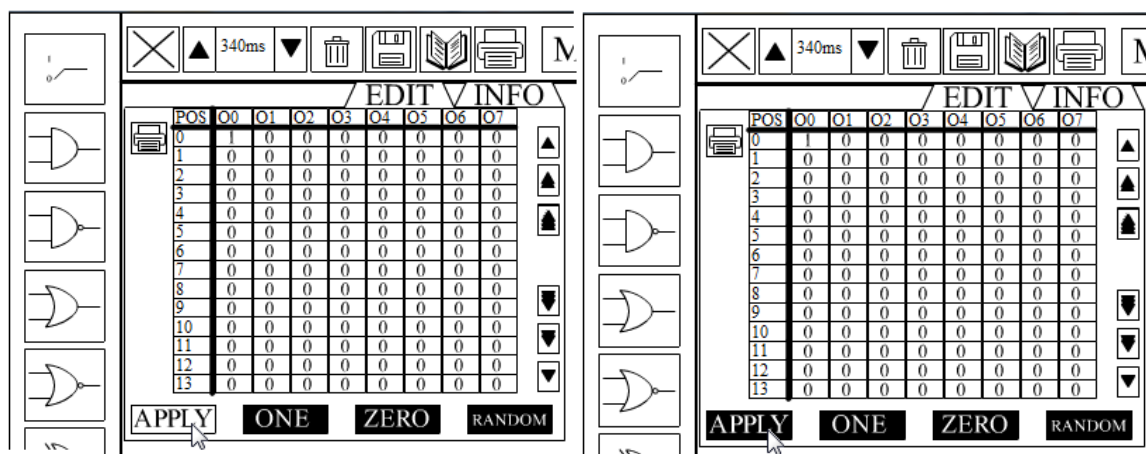


Figura 4.116 – Validação das edições na tabela.

O botão com imagem de uma impressora é responsável por realizar a impressão do conteúdo da memória. Quando esse botão é selecionado, a janela de impressão do sistema operacional é chamada. Caso a impressão seja confirmada na janela, uma folha será impressa contendo a informações da memória disposta em colunas que contém no máximo noventa e quatro posições cada. Para a memória com o maior numero de posições, mil e vinte e quatro

posições, a folha contém onze colunas. Para que o processo de impressão seja executado é necessário que as modificações na tabela sejam validadas pela seleção do botão APPLY.

POS	00000000	01234567
0000:	10000000	
0001:	01000000	
0002:	00100000	
0003:	00010000	
0004:	00001000	
0005:	00000000	
0006:	00000000	
0007:	00000000	

Figura 4.117 – Folha de impressão das posições da memória.

A Figura 4.117 ilustra a folha resultante do processo de impressão das posições da memória. Nesse caso foi utilizada uma memória com oito posições e oitos dígitos. As posições da memória ficam dispostas abaixo de um cabeçalho que indica a coluna referente ao número de identificação da posição da memória, e as colunas referentes aos dígitos de cada posição da memória. Caso a memória possua um número elevado de posições, mais de um cabeçalho pode ser utilizado. Assim que as noventa e quatro posições disponíveis para o primeiro cabeçalho forem preenchidas, um segundo cabeçalho é criado à direita deste, e as posições seguintes são inseridas abaixo do segundo cabeçalho. Esse processo pode se repetir até que todas as posições sejam inseridas na folha de impressão.

Para encerra o editor deve-se selecionar o botão INICIAR/CANCELAR, que exibe a imagem de um "X", assim todas as modificações na aba "EDIT" não validadas pelo botão APPLY são perdidas, as modificações na aba "INFO" são aplicadas à porta lógica e a janela do editor é retirada da Área de Trabalho. O botão INICIAR/CANCELAR volta a exibir sua imagem original.

4.6.2. Editor de Minimização

O Editor de Minimização é uma ferramenta que tem a função de realizar a tradução de uma Tabela Verdade em uma função booleana minimizada e também tem uma segunda função que é de informar a função booleana minimizada correspondente a um determinado trecho de um circuito selecionado.

Para se iniciar o editor deve-se selecionar o botão MINIMIZAR, assim a segunda função do editor é iniciada, o botão MINIMIZAR passa a exibir uma imagem que indica que o editor está ativo e o ícone INICIAR/CANCELAR passa a exibir a imagem de um "X". A ativação desse editor só pode ser feita enquanto o programa estiver com nenhuma função e editor ativos, ou quando o Editor de Macros estiver ativo, ou quando uma das funções de navegação, Arrasto, Ampliar e Reduzir, estiverem ativas sozinhas.

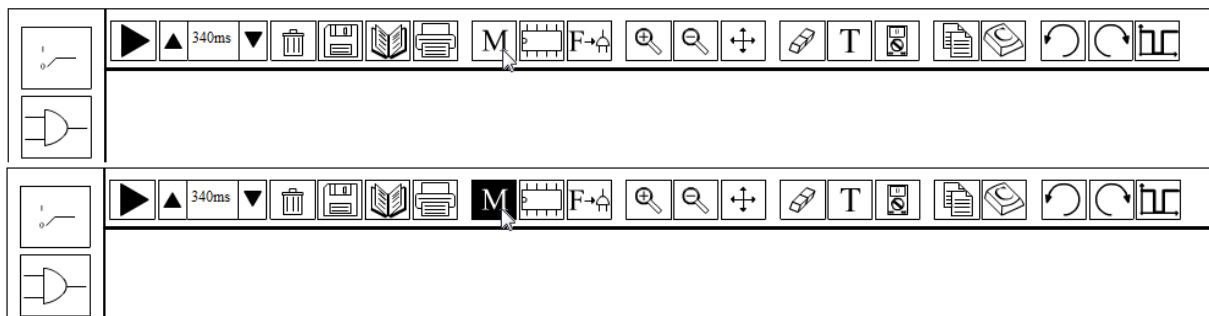


Figura 4.118 – Ativação do Editor de Minimização.

Após iniciar o editor, a função ativa é a responsável por realizar a análise de um circuito e exibir uma função booleana minimizada correspondente a esse circuito. Para realizar essa análise deve-se primeiro selecionar chaves lógicas e uma porta lógica de tipo Led. As chaves lógicas representam as variáveis de entrada da função e a porta lógica de tipo Led representa a saída da função. O processo de seleção das portas lógicas é feito ao se posicionar o cursor do mouse sobre as chaves lógicas e pressionar o botão esquerdo do mouse, nesse momento a chave lógica selecionada tem a cor de sua imagem alterada para verde e sua caixa de texto passa a exibir, no lugar de seu nome, a letra correspondente a variável que essa chave lógica representará. Como há um limite de oito variáveis de entradas para esse editor, o número máximo de chaves lógicas selecionadas é oito, e as variáveis de entradas são "A", "B", "C", "D", "E", "F", "G" e "H". Durante o processo de seleção das portas lógicas, é possível navegar para a Área de Trabalho utilizando as funções de Arrasto, Ampliar e Reduzir.

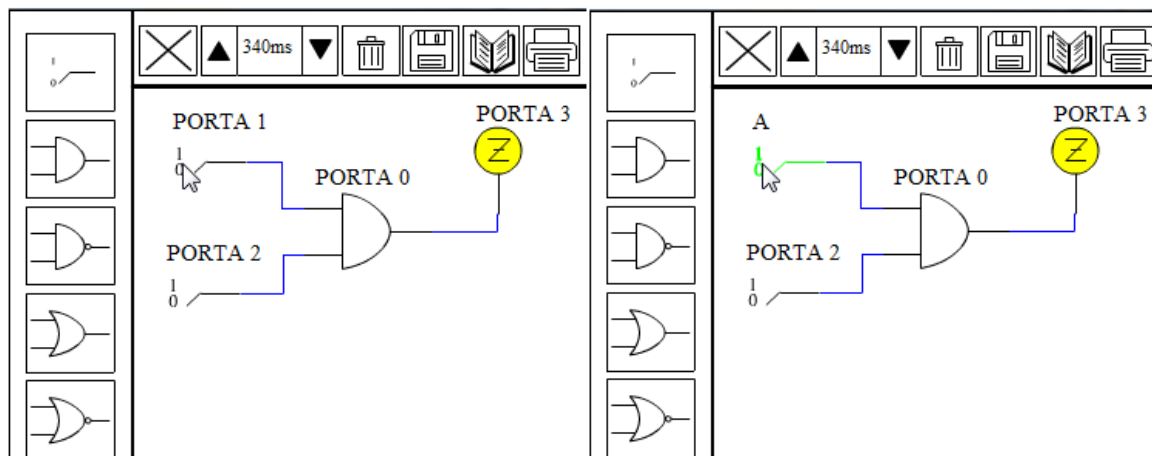


Figura 4.119 – Seleção de chave lógica.

O processo de seleção da porta lógica de tipo Led é o mesmo das chaves lógicas, porém somente uma porta lógica de tipo Led pode ser selecionada e em sua caixa de texto passa a ser exibida a letra "S", que representa a saída da função.

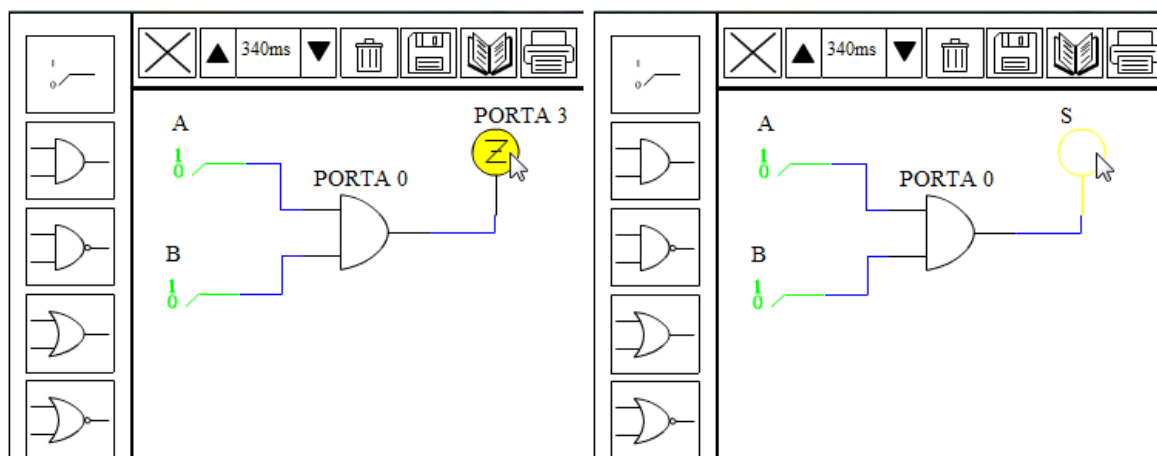


Figura 4.120 – Seleção de porta lógica de tipo Led.

O circuito analisado pelo editor é aquele que está compreendido entre as chaves lógicas e a porta lógica de tipo Led, que representam as entradas e a saídas da função que será criada. Assim, pode-se definir o circuito analisado como sendo o circuito compreendido entre as entradas e a saída da função. Entende-se como circuito compreendido entre as entradas e a saída da função, o grupo de portas lógicas que, através de suas ligações, tem contato direto ou indireto com as portas lógicas selecionadas para executar o editor. A definição de contato direto entre portas lógicas pode ser descrita como a situação em que duas portas lógicas possuem um ou mais fios ligando seus pinos, e a definição de contato indireto entre portas lógicas pode ser descrita como a situação em que duas portas lógicas não possuem fios ligando seus pinos, porém estão ligadas por fios a outras portas lógicas, e essas por sua vez

também estão ligadas a outras portas lógicas, e assim sucessivamente até que, partindo de qualquer uma das duas portas lógicas originais e seguindo as ligações entre as portas lógicas é possível alcançar a outra porta lógica original.

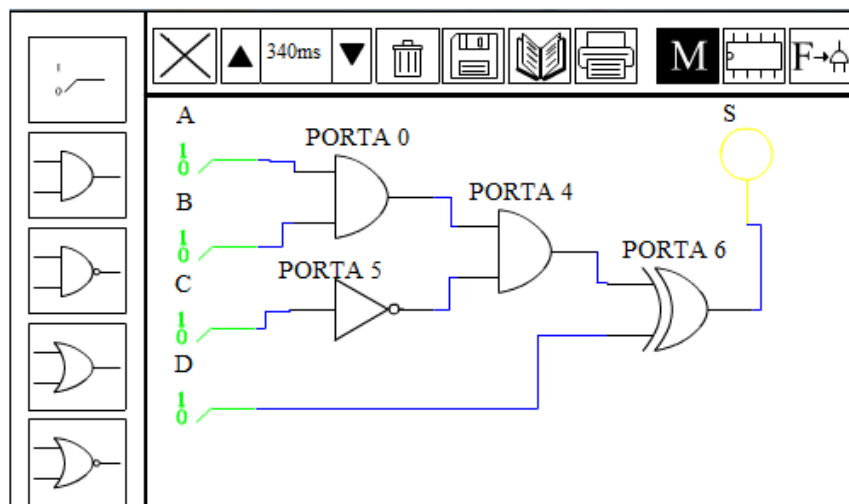


Figura 4.121 – Circuito analisado pelo Editor de Minimização.

Na Figura 4.121 é possível notar que a porta lógica "PORTA 4" não está diretamente ligada a nenhuma das portas lógicas que representam as entradas e a saída da função, porém como ela está ligada às portas lógicas "PORTA 0", "PORTA 5" e "PORTA 6", e essas possuem ligação direta com as portas lógicas que representam entradas e a saída da função, a porta lógica "PORTA 4" também será incluída no circuito analisado, pois nesse caso constitui-se ligação indireta entre essa porta lógica e as portas lógicas que representam as entradas e a saída da função.

Caso existam outros circuitos na Área de Trabalho externos ao circuito compreendido entre as chaves lógicas e a porta lógica de tipo Led, ou seja, um conjunto de portas lógicas que não possua ligação direta ou indireta com as portas lógicas que representam as entradas e a saída da função, estas portas lógicas não são levados em consideração no processo da análise.

Após selecionar as chaves lógicas e uma porta lógica de tipo Led, o editor pode ser executado para que possa informar qual é a função booleana que representa o circuito analisado, assim deve-se selecionar o botão MINIMIZAR, que apresenta a imagem alterada. Nesse momento o editor realiza as análises e informa na caixa de texto da porta lógica de tipo Led a função booleana minimizada após a letra "S". É necessário que pelo menos uma chave e

exatamente uma porta lógica de tipo Led sejam selecionadas para que o editor possa ser executado, caso contrário o processo descrito acima não executa o editor.

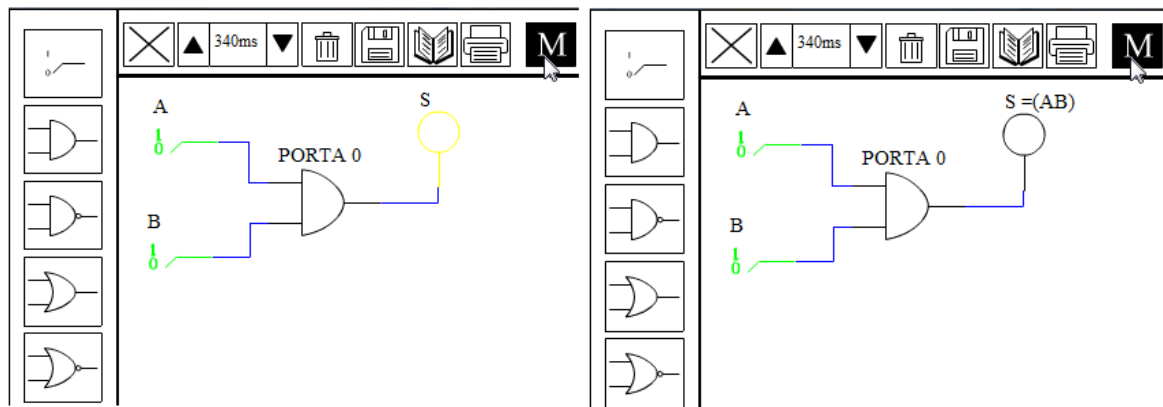


Figura 4.122 – Resultado da execução do editor de minimização.

A Figura 4.122 ilustra a execução do editor, e neste caso o circuito compreendido entre as entradas e a saída da função é formado por uma porta lógica de tipo AND, e assim a caixa de texto da porta lógica de tipo Led foi atualizada e passa a exibir o texto “ $S=(AB)$ ” que representa o circuito analisado de maneira correta.

Caso existam chaves lógicas não selecionadas ligadas ao circuito compreendido entre as entradas e a saída da função, a análise do circuito será feita mantendo essas chaves com seu estado lógico de saída inalterado, pois estas não são consideradas entradas da função e dessa maneira podem ser consideradas portas lógicas de tipo VCC ou GND, caso tenham suas saída em estado lógico um ou zero respectivamente.

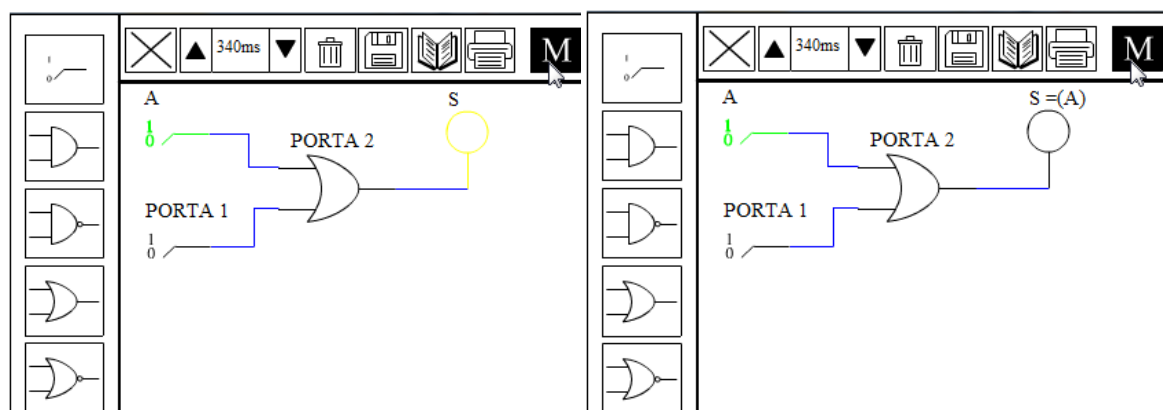


Figura 4.123 – Resultado da execução do editor de minimização, com chave lógica não selecionada.

Existe uma situação em que o editor não consegue analisar o circuito, e essa situação ocorre quando o circuito analisado contém alguma porta lógica de tipo Pulser, ou alguma

configuração de ligação entre os componentes do circuito que gerem realimentações instáveis. Em ambos os casos o editor informa através da barra de informações que o circuito é instável e é impossível encontrar uma função que o represente.

A análise de circuitos que contenham portas lógicas de tipo Flip-Flop, ou ligações entre portas lógicas que reproduzam características desse tipo de porta lógica, possui grande probabilidade de apresentar resultados errados, pois a natureza sequencial desse tipo de portas lógicas não pode ser traduzida em funções booleanas simples. Assim apenas circuitos com características puramente combinacionais apresentam resultados corretos sempre.

Após executar o editor e obter a função booleana do circuito analisado, ou mesmo quando a execução do editor resulta no aviso de sistema instável, o editor entra em um estado onde não é possível selecionar novas portas lógicas e também não é possível executar o editor novamente, apenas é possível navegar para a Área de Trabalho utilizando as funções de Arrasto, Ampliar e Reduzir. Nesse caso deve-se selecionar o botão INICIAR/CANCELAR, que exibe a imagem de um "X", para encerrar o editor. Nesse momento as caixas de texto das portas lógicas selecionadas voltam a exibir seus nomes e são retiradas da Área de Trabalho. Caso não estivessem sendo exibidas antes de serem selecionadas, as portas lógicas selecionadas voltam à cor original, o botão MINIMIZAR volta a exibir sua imagem original e o botão INICIAR/CANCELAR volta a exibir sua imagem original. O processo de encerramento do editor pode ser executado também antes de se executar o editor.

Não é possível cancelar a seleção de uma porta lógica que já tenha sido selecionada antes da execução do editor. Caso uma porta lógica tenha sido selecionada de maneira errada, deve-se encerrar o editor e reiniciá-lo, assim será possível selecionar as portas lógicas novamente.

Caso o botão MINIMIZAR seja selecionado após a inicialização do editor e com nenhuma porta lógica selecionada, o editor encerra o seu funcionamento como analisador de circuitos e passa a exibir uma tabela na Área de Trabalho, indicando que seu funcionamento passou a ser o de realizar a tradução da Tabela Verdade em uma função booleana minimizada.

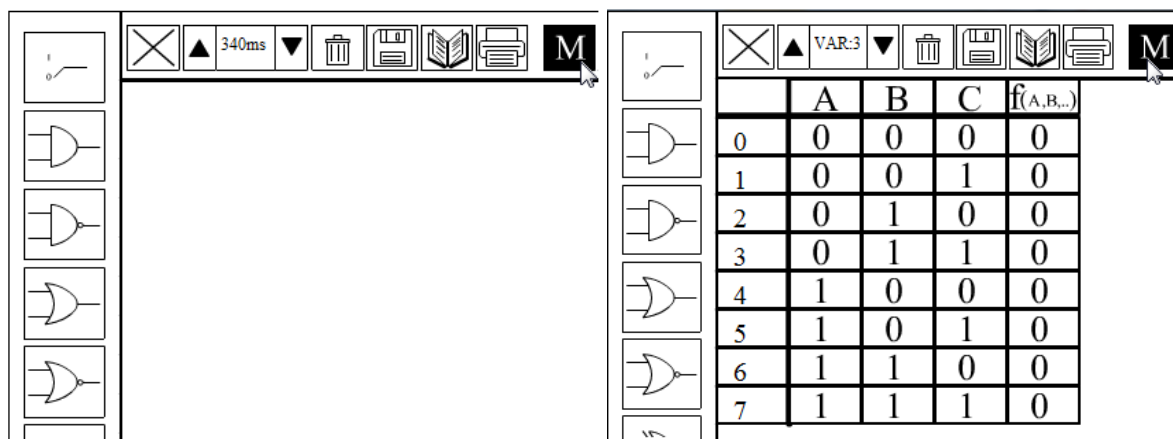


Figura 4.124 – Ativação da tabela do editor de minimização.

A tabela exibida assim como ilustrado na Figura 4.124, possui uma coluna de cabeçalho contendo o nome das variáveis de entradas nas primeiras células e na última célula um indicativo da função. A tabela inicia com três variáveis de entrada nomeadas "A", "B" e "C", assim são necessárias oito linhas para cobrir todas as possíveis combinações de estados dessas três variáveis, totalizando nove linhas na tabela.

Com a tabela inserida na Área de Trabalho, deve-se inserir nela a informação de quais estados lógicos cada posição da coluna correspondente a função de saída deve ter. Para isso basta posicionar o cursor do mouse sobre a célula, na última coluna da tabela, que esteja na linha da posição desejada e pressionar o botão esquerdo do mouse, assim a célula deixa de exibir a imagem "0" e passa a exibir a imagem "1". Caso o processo descrito seja repetido na mesma célula, ela deixa de exibir a imagem "1" e passa a exibir a imagem "X", e caso o processo volte a ser repetido, passa a exibir a imagem "0", voltando ao início do ciclo "0", "1" e "X". As imagens exibidas correspondem ao estado lógico zero, um e don't care.

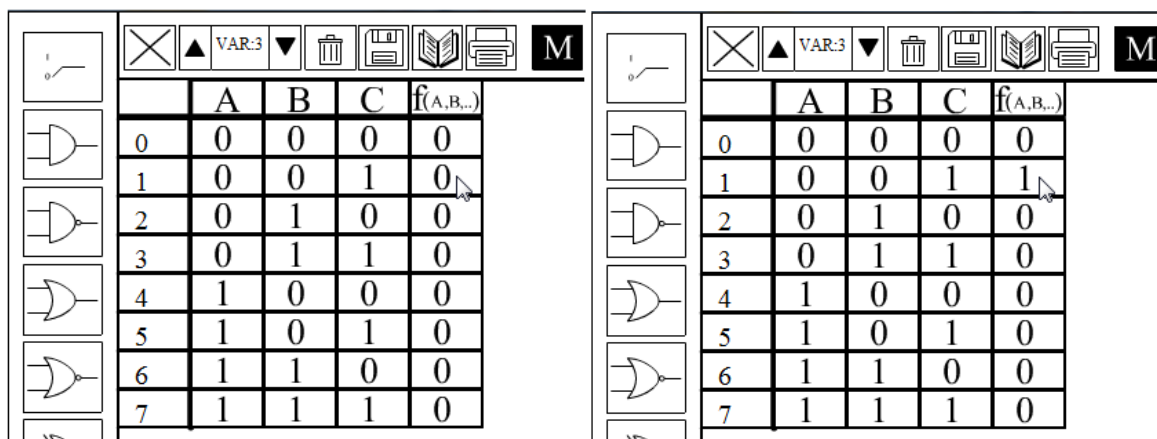


Figura 4.125 – Edição do estado lógico da célula da tabela.

Após preencher a última coluna da tabela com as informações do estado lógico da saída em cada posição da tabela, deve-se selecionar o botão MINIMIZAR para executar o editor. Nesse momento o editor analisará as informações contidas na tabela e informará na Barra de Informações a função booleana correspondente à tabela. A função permanece na Barra de Informações, até que o editor seja executado novamente ou até que o editor seja encerrado.

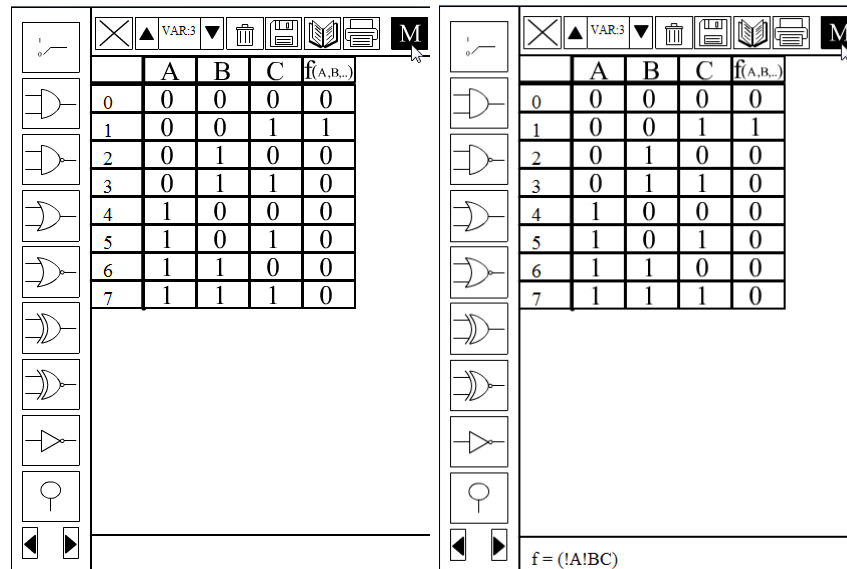


Figura 4.126 – Execução do editor de minimização.

O mostrador da Barra de Ferramentas deixa de exibir o valor do período de atualização da execução da simulação e passa a exibir o número de variáveis de entrada da tabela. As setas adjacentes ao mostrador, direcionadas para baixo e para cima, após serem selecionadas passam a editar o número de variáveis da tabela, retirando e adicionando uma unidade ao número de variáveis da tabela. No momento em que essa edição é realizada, a tabela é redimensionada para comportar o novo número de variáveis de entrada. O valor mínimo de variáveis na tabela é dois e o máximo é oito. As edições no número de variáveis reiniciam a tabela por completo e todas as edições nas células são descartadas.

	A	B	C	f(A,B,C)
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

	A	B	C	D	f(A,B,C,D)
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	0

Figura 4.127 – Edição no número de variáveis da tabela.

Para tabelas com número de variáveis superior a quatro, o número máximo de linhas exibidas é dezesseis e são inseridos botões de navegação para editar o trecho da tabela exibido na janela do programa. Estes botões são divididos em dois grupos, um grupo de três botões que fazem com que posições de menor número de identificação sejam exibidos e um grupo de três botões que fazem com que posições de maior número de identificação sejam exibidos.

	A	B	C	D	E	f(A,B,C,D,E)
0	0	0	0	0	0	0
1	0	0	0	0	1	0
2	0	0	0	1	0	0
3	0	0	0	1	1	0
4	0	0	1	0	0	0
5	0	0	1	0	1	0
6	0	0	1	1	0	0
7	0	0	1	1	1	0
8	0	1	0	0	0	0
9	0	1	0	0	1	0
10	0	1	0	1	0	0
11	0	1	0	1	1	0
12	0	1	1	0	0	0
13	0	1	1	0	1	0
14	0	1	1	1	0	0
15	0	1	1	1	1	0

Figura 4.128 – Tabela com cinco variáveis.

Na tabela da Figura 4.128 as dezesseis primeiras posições estão sendo exibidas inicialmente, e ao selecionar o botão com uma seta direcionada para baixo, a primeira linha passa a exibir a segunda posição da tabela e a décima sexta linha passa a exibir a décima sétima posição. Assim ao pressionar esse botão, cada linha passa a exibir a posição da tabela

de número de identificação igual à posição anteriormente exibida adicionada de uma unidade. Caso os botões com duas ou três setas direcionadas para baixo tivessem sido selecionados, seriam adicionadas dez e cem unidades, respectivamente.

	A	B	C	D	E	f(A,B,...)
0	0	0	0	0	0	0
1	0	0	0	0	1	0
2	0	0	0	1	0	0
3	0	0	0	1	1	0
4	0	0	1	0	0	0
5	0	0	1	0	1	0
6	0	0	1	1	0	0
7	0	0	1	1	1	0
8	0	1	0	0	0	0
9	0	1	0	0	1	0
10	0	1	0	1	0	0
11	0	1	0	1	1	0
12	0	1	1	0	0	0
13	0	1	1	0	1	0
14	0	1	1	1	0	0
15	0	1	1	1	1	0

	A	B	C	D	E	f(A,B,...)
10	0	1	0	1	0	0
11	0	1	0	1	1	0
12	0	1	1	0	0	0
13	0	1	1	0	1	0
14	0	1	1	1	0	0
15	0	1	1	1	1	0
16	1	0	0	0	0	0
17	1	0	0	0	1	0
18	1	0	0	1	0	0
19	1	0	0	1	1	0
20	1	0	1	0	0	0
21	1	0	1	0	1	0
22	1	0	1	1	0	0
23	1	0	1	1	1	0
24	1	1	0	0	0	0
25	1	1	0	0	1	0

Figura 4.129 – Navegação pelas posições da tabela.

Caso os botões com uma, duas ou três setas direcionadas para cima forem selecionados, cada linha passa a exibir a posição da tabela de número de identificação igual à posição anteriormente exibida subtraída de uma, dez e cem unidades, respectivamente.

Caso a seleção de algum botão resulte na exibição de posições não existentes, a navegação para essas posições é interrompida e as posições iniciais serão exibidas, no caso de botões com setas direcionadas para cima, e as últimas posições serão exibidas no caso de botões com setas direcionadas para baixo.

A qualquer momento enquanto a tabela é exibida o editor pode ser encerrado, selecionando o botão INICIAR/CANCELAR, que exibe a imagem de um "X". Nesse momento o editor é encerrado, a Barra de Informações volta a exibir informações sobre o programa, a tabela é retirada da Área de Trabalho, o botão MINIMIZAR volta a exibir sua imagem original e o botão INICIAR/CANCELAR volta a exibir sua imagem original.

O Editor de Minimização possui integração com o Editor de Interpretação. Essa integração só existe quando o Editor de Minimização está com a tabela sendo exibida, e é feita quando, tendo o resultado da minimização exibido na barra de informações, o botão INTERPRETAR for selecionado. Nesse momento o Editor de Minimização é encerrado e a

informação da função que era exibida na Barra de Informações é passada ao Editor de Interpretação, que é inicializado e executado com a informação que foi passada.

Todo o código desse editor foi feito tomando como base o Algoritmo de Quine McCluskey, que é um método projetado para encontrar os primos implicantes de uma função booleana, e realizar a seleção do conjunto de primos implicantes que cubram todas as saídas com nível lógico um da função. Após a seleção do conjunto de primos implicantes, a função é montada.

As etapas do Algoritmo de Quine McCluskey são explicadas no Anexo 1.

4.6.3. Editor de Interpretação de Funções

O Editor de Interpretação de Funções realiza a tradução de uma função booleana em um circuito lógico. O editor realiza uma série de análises sobre a função booleana buscando por erros de sintaxe e apenas se nenhum erro for encontrado o circuito é criado. Para se iniciar o editor deve-se selecionar o botão INTERPRETAR, assim a janela do editor é inserida na Área de Trabalho e o botão INTERPRETAR passa a exibir imagem que indica que o editor está ativo.

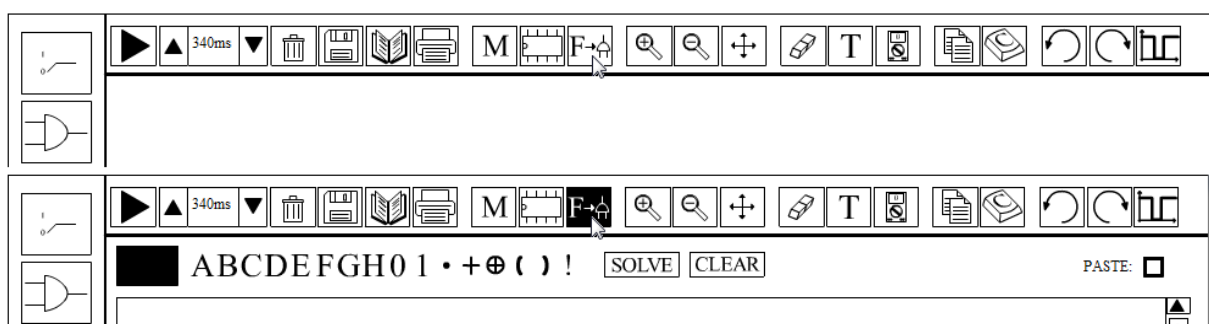


Figura 4.130 – Inicialização do editor de interpretação de funções.

A janela do editor é dividida em três regiões. A primeira região é posicionada abaixo da Barra de Ferramentas, e nela estão contidos um retângulo de cor preta de exclusão, o conjunto de variáveis e operadores, os botões SOLVE e CLEAR, e o marcador PASTE. A segunda região encontra-se abaixo da primeira e contém três caixas de texto alinhadas e à direita um conjunto de botões e mostrador. A terceira região encontra-se próxima à base da janela do

editor e é composta por uma área de trabalho do editor e nela encontram-se botões de navegação.

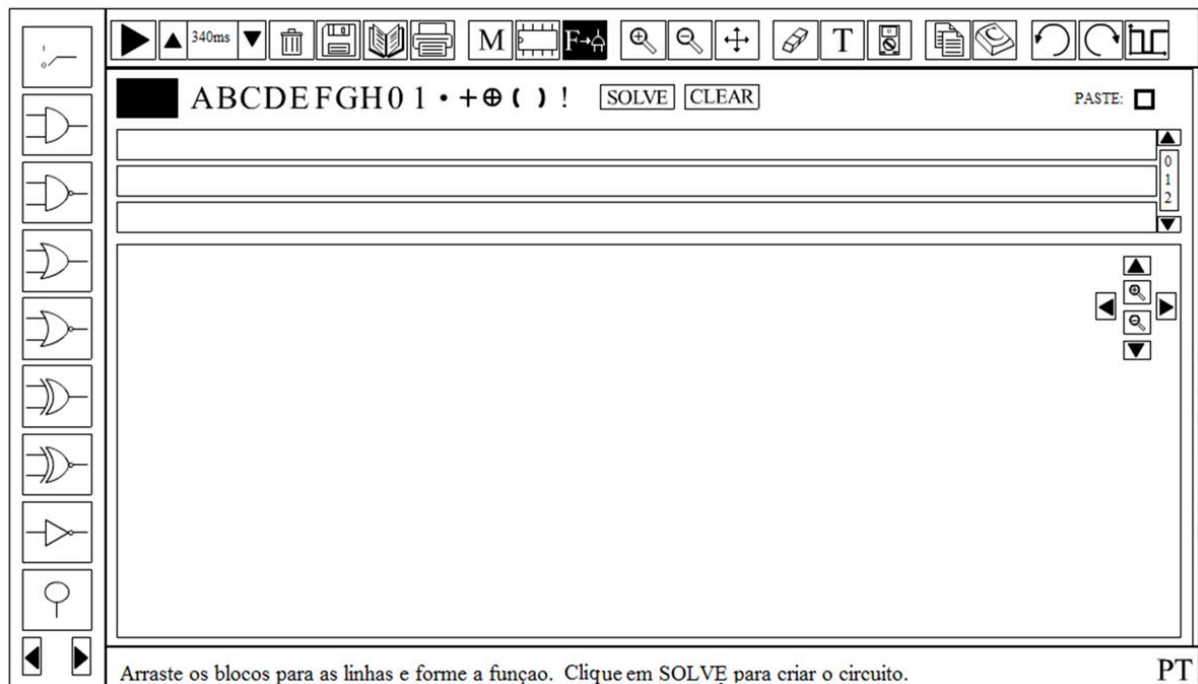


Figura 4.131 – Janela do Editor de Interpretação de Funções.

Para montar a função que se deseja obter o circuito, deve-se posicionar o cursor do mouse sobre uma das variáveis ou operadores que se encontram na primeira região da janela e pressionar e soltar o botão esquerdo do mouse. Nesse momento será criada uma cópia da variável ou operador selecionado, e essa cópia acompanha o movimento do mouse.

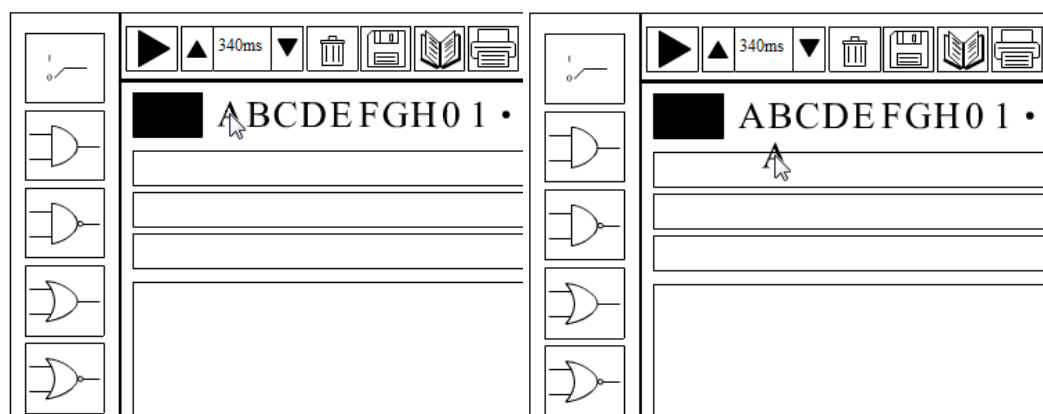


Figura 4.132 – Criação da variável A.

Depois de criada a cópia da variável como é ilustrado na Figura 4.132, deve-se posicionar o mouse sobre uma das três caixas de texto presentes na segunda região, e pressionar o botão esquerdo do mouse. Assim a cópia da variável será inserida na caixa de texto e deixa de

acompanhar o movimento do mouse. Esse processo deve ser repetido para todas as variáveis e operadores que irão formar a função.

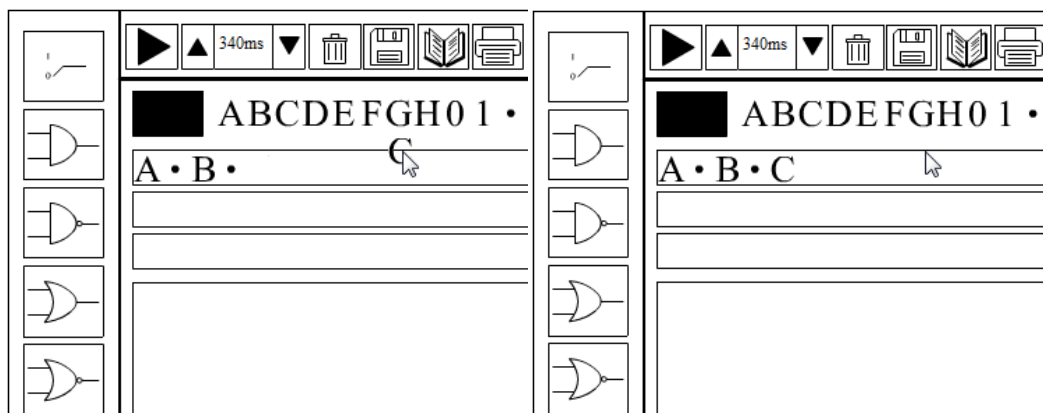


Figura 4.133 – Inserção da função na caixa de texto.

É permitido realizar o processo de inserção de variáveis na caixa de texto posicionando o cursor do mouse em qualquer posição no interior da caixa de texto. Como é ilustrado na Figura 4.133, o cursor do mouse se encontra em uma posição à direita das variáveis e operadores já inseridos na caixa de texto assim, ao finalizar o processo de inserção, a variável se posiciona à direita das variáveis e operadores já inseridos, ou seja, há um empilhamento à esquerda das variáveis e operadores inseridos na caixa de texto. Porém, se o cursor do mouse for posicionado em uma região da caixa de texto onde já existam outras variáveis ou operadores inseridos, estes são deslocados e deixam um espaço vazio na região onde o cursor do mouse se encontra, e esse espaço é onde a nova variável ou operador será inserido caso o processo de inserção seja concluído. O processo descrito é repetido caso o cursor do mouse seja movimentado sobre um conjunto de variáveis e operadores, o conjunto se desloca para mostrar qual será a posição em que a nova variável ou operador será inserido.

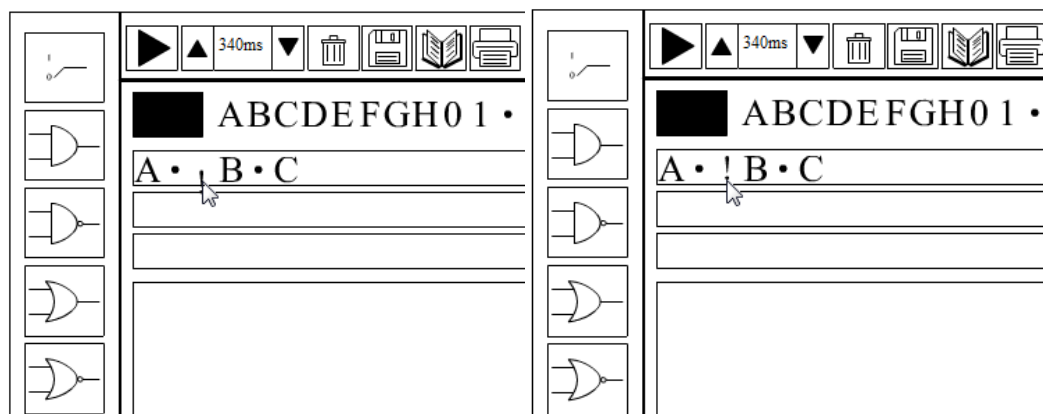


Figura 4.134 – Inserção do operador ! com deslocamento das demais variáveis e operadores.

Caso o cursor do mouse seja movimentado para uma posição em que deixe de estar sobre a caixa de texto, e considerando que estivesse ocorrendo uma situação de deslocamento como ilustrado na Figura 4.134, esse deslocamento é desfeito e as variáveis e operadores voltam a ocupar suas posições originais.

Caso a função desejada seja muito grande, pode ocorrer a situação de ela não caber em uma só caixa de texto. Neste caso, deve-se inserir as variáveis e operadores nas demais caixas de texto. A ordem de leitura da função segue o padrão de leitura normal de um texto em português, assim a leitura se inicia da esquerda para a direita e ao final da caixa de texto, a leitura reinicia na esquerda da caixa de texto imediatamente abaixo. Não é necessário preencher completamente uma caixa de texto para iniciar o preenchimento de outra caixa, e também é permitido deixar caixas de texto vazias entre caixas não vazias. O editor segue a ordem de leitura da função ignorando as caixas vazias.

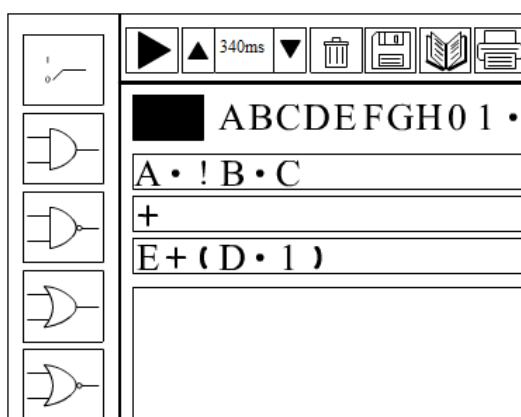


Figura 4.135 – Caixas de texto preenchidas.

Na janela do editor são exibidas somente três caixas de texto por vez, porém estão disponíveis cem caixas de texto. O mostrador da segunda região mostra o número de identificação das caixas de texto que são exibidas na janela. Para mudar as caixas de textos que são exibidas deve-se posicionar o cursor do mouse sobre um dos botões junto ao mostrador, pressionar e soltar o botão esquerdo do mouse. O botão com a imagem de uma seta direcionada para baixo, quando selecionado, faz com que as novas caixas de texto exibidas na janela tenham número de identificação igual ao número de identificação das caixas de texto que eram exibidas anteriormente, adicionados de uma unidade. O mostrador passa a exibir os novos números de identificação das caixas de texto.

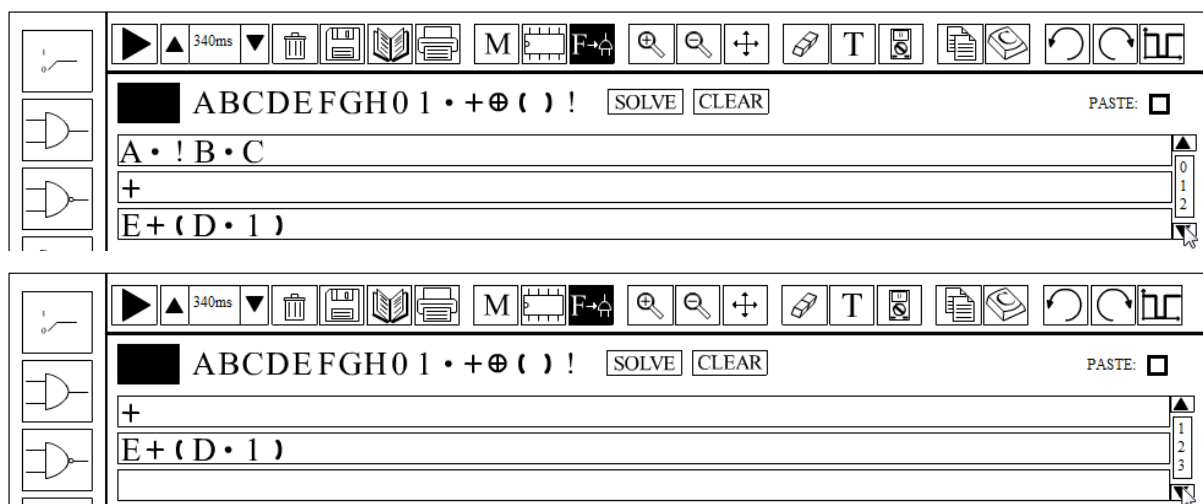


Figura 4.136 – Mudança das caixas de texto exibidas.

O mesmo processo ilustrado na Figura 4.136 ocorre caso o botão com a imagem da seta direcionada para cima seja selecionado, porém nesse caso ocorre a subtração de uma unidade ao número de identificação das caixas de texto anteriormente exibidas. Caso o mostrador exiba as três primeiras posições, a seleção do botão com imagem da seta direcionada para cima não tem efeito, e no caso de o mostrador exibir as três últimas posições, a seleção do botão com imagem da seta direcionada para baixo também não tem efeito.

Caso alguma variável ou operador esteja em local inapropriado e deseje-se reposicioná-lo, deve-se posicionar o cursor do mouse sobre esta variável ou operador, e pressionar o botão esquerdo do mouse. Nesse momento a variável ou operador passa a acompanhar o movimento do mouse e é retirado da caixa de texto.

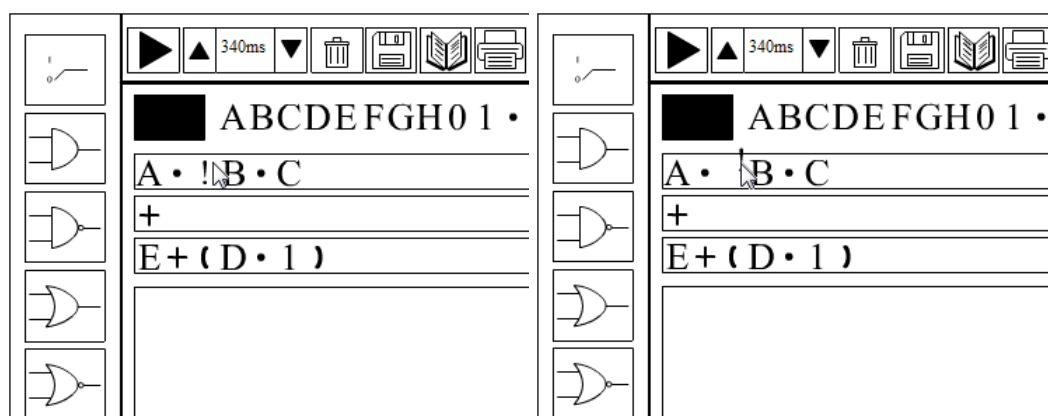


Figura 4.137 – Seleção da variável inserida.

Após esse procedimento, duas ações podem ser tomadas, uma é a exclusão da variável ou operador, ou o reposicionamento dentro de alguma caixa de texto. Para realizar a exclusão

deve-se posicionar o cursor do mouse sobre o retângulo de cor preta na primeira região da janela e pressionar o botão esquerdo do mouse.

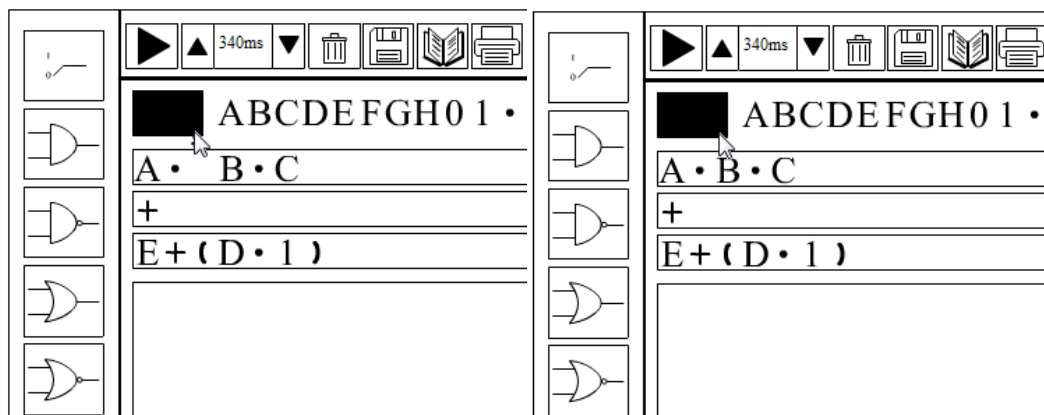


Figura 4.138 – Exclusão de variável.

Para realizar o reposicionamento deve-se proceder como uma nova inserção, não havendo a necessidade de manter a variável ou operador na mesma caixa de texto em que estavam.

Após inserir a função desejada na caixa de texto, deve-se posicionar o cursor do mouse sobre o botão SOLVE da primeira região da janela e pressionar o botão esquerdo do mouse para que o editor possa ser executado. Após executar o editor, este realiza uma série de análises na função criada em busca de erros de sintaxe e caso não encontre nenhum tipo de erro, o circuito referente à função é criado e inserido na terceira região da janela.

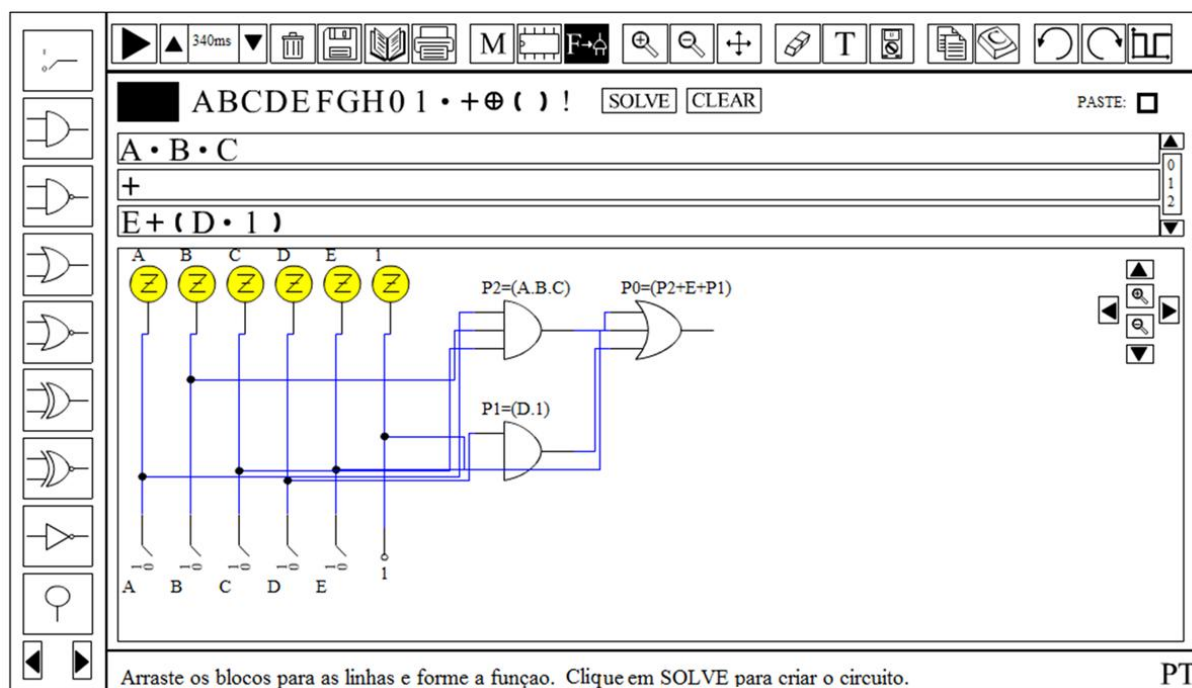


Figura 4.139 – Execução do editor cria o circuito.

Os erros de sintaxe que o editor procura são: variáveis adjacentes sem operador entre elas, número errado de parênteses, posicionamento errado de operadores dentro de parênteses, parênteses que não contêm operação entre variáveis.

A análise da função é feita pela identificação dos parênteses que foram inseridos e também pela identificação de quais operações dentro da função tem prioridade sobre as demais. A operação de produto lógico é a operação que tem a maior prioridade, seguida da operação de soma lógica exclusiva, e com a operação de soma lógica como a operação de menor prioridade. Assim, em uma função onde não existam parênteses, o editor isola primeiro as operações de produto lógico em parênteses, em seguida o editor isola em parênteses as operações de soma lógica exclusiva, e por último isola as operações de soma lógicas em parênteses. Depois de identificar qual é a ordem em que as operações na função devem ser realizadas por meio do isolamento das operações em parênteses, o editor cria as portas lógicas que corresponde a cada operação dos parênteses, e realiza as ligações entre essas portas lógicas conforme as operações entre os parênteses às quais essas portas lógicas representam.

Após a criação das portas lógicas, o editor insere no nome de cada uma delas a identificação de qual parêntese ela representa, letra "P" seguida do número de identificação desse parêntese, seguido da operação que esse parênteses representa. Os parênteses que contenham operações envolvendo outros parênteses, como no caso da porta lógica "P0" na Figura 4.139, apresentam o nome desses outros parênteses em sua caixa de texto substituindo a operação que representa. Assim o nome da porta lógica "P0" é seguida pela operação " $=P2+E+P1$ ", onde "P2" e "P1" são os números de identificação das porta lógica cuja saídas fazem parte da soma lógica com a variável E.

O circuito criado, além de conter as portas lógicas que traduzem a função booleana, também contém um conjunto de portas lógicas de tipo Led, chaves lógicas e portas de tipo VCC e GND. Essas portas lógicas representam as variáveis de entrada presentes na função e são ligadas às portas lógicas do circuito. Caso alguma variável de entrada tenha sido negada na função, além da chave lógica representando essa variável, uma porta lógica de tipo NOT é inserida e ligada à saída dessa chave lógica, e sua saída é ligada ao restante do circuito. O editor consegue identificar casos de duas ou mais negações consecutivas sobre a mesma variável, e somente insere a porta lógica de tipo NOT caso haja número ímpar de negações sucessivas sobre uma variável.

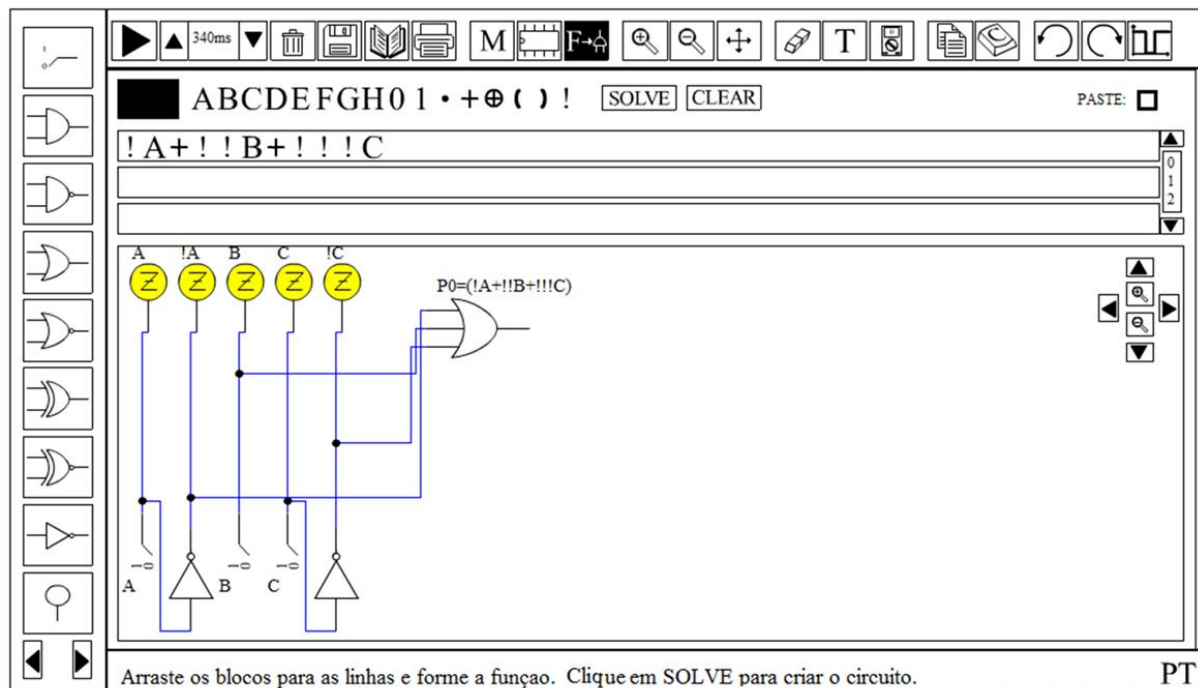


Figura 4.140 – Função com variáveis negadas.

A negação de variáveis deve sempre ser feita pela inserção do operador "!" imediatamente à esquerda da variável que se deseja negar. Para negações sucessivas deve-se inserir mais operadores "!" à esquerda da porta lógica, com a exigência que eles sejam adjacentes.

Caso o circuito criado seja grande demais para ser exibido totalmente na área de trabalho da janela do editor, é possível navegar pela área de trabalho da janela utilizando os botões de navegação dispostos próximos ao seu canto superior direito. Os botões com setas direcionadas para direita, esquerda, baixo e cima quando selecionados provocam o arrasto da área de trabalho da janela, e só estão liberados para serem operados quando o circuito não pode ser exibido em sua totalidade na área de trabalho da janela. Os botões de AMPLIAR e REDUZIR realizam, respectivamente, a ampliação e redução das dimensões da área de trabalho da janela, e podem ser operados para qualquer configuração de circuito.

O botão CLEAR da primeira região da janela é responsável por apagar a função inserida nas caixas de texto e apagar o circuito criado na área de trabalho da janela.

Quando o editor é iniciado, o marcador PASTE está desmarcado, e para marca-lo deve-se posicionar o cursor do mouse sobre ele e pressionar o botão esquerdo do mouse, nesse momento ele passa a exibir a imagem de um quadrado preenchido de preto e a opção de colar o circuito criado no editor na Área de Trabalho é ativada.

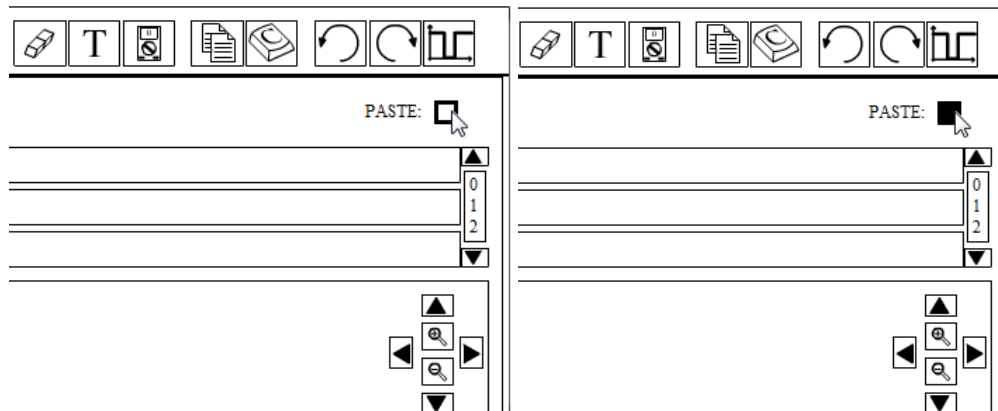


Figura 4.141 – Ativação do marcador.

Para encerrar o editor deve-se selecionar o botão. Nesse momento a janela do editor é retirada da Área de Trabalho e o botão INTERPRETAR volta a exibir sua imagem original. Caso o marcador PASTE esteja ativado no momento em que o editor for encerrado e caso haja algum circuito criado na área de trabalho da janela, o editor copia esse circuito para a Área de Trabalho do programa e pede que o usuário indique a posição desse grupo.

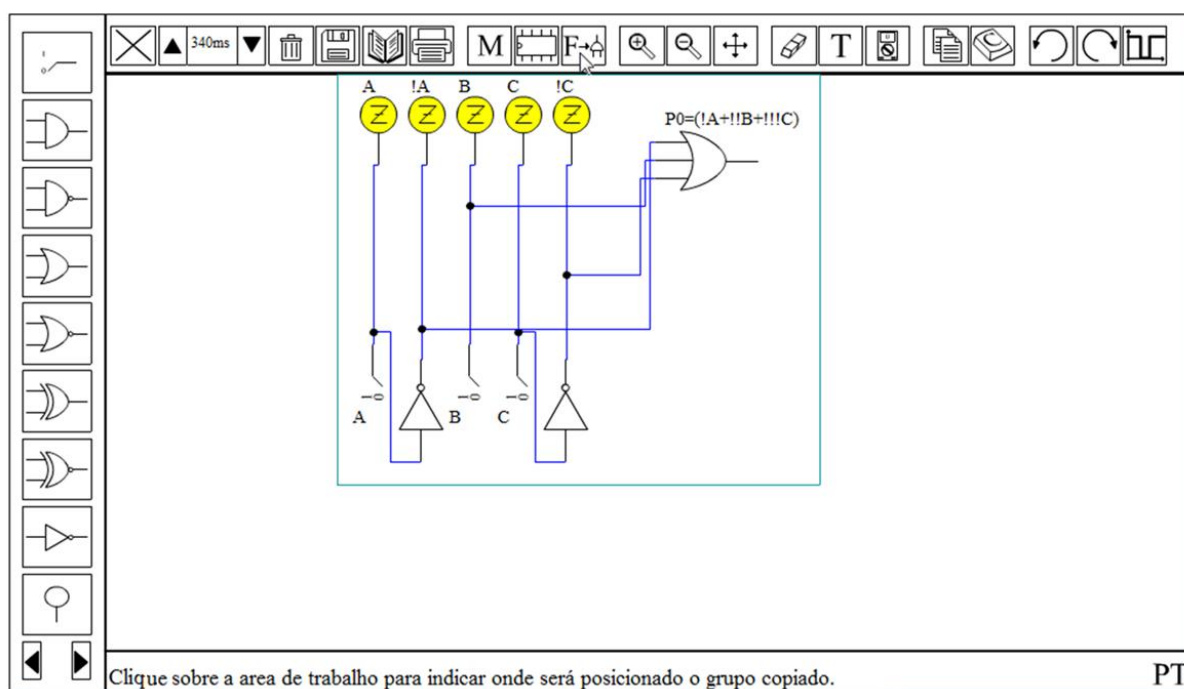


Figura 4.142 – Inserção do circuito do editor na Área de Trabalho.

O procedimento de inserção do circuito ilustrado na Figura 4.142 segue as mesmas regras apresentadas na função PASTE, uma vez que esse procedimento representa uma situação de duplicação de circuito.

A integração do Editor de Interpretação de Funções com o Editor de Minimização foi descrito no capítulo 4.6.2 e descreve que quando o Editor de interpretação de Funções é inicializado durante a execução do Editor de Minimização com a tabela sendo exibida, a janela do Editor de Interpretação de Funções é exibida, e nele o circuito equivalente à função que era exibida na Barra de Informações está criado em sua área de trabalho e só é possível navegar pela área de trabalho do editor, fechar o editor realizando ou não a duplicação do circuito criado, ou selecionar o botão CLEAR para reiniciar o editor. Existe uma restrição a essa integração que é o caso em que a função resultante do Editor de Minimização não apresentar nenhuma operação, apenas valor lógico um e zero, ou apenas uma única variável. Nesse caso um circuito não pode ser criado e o Editor de Interpretação de Funções inicializa sem exibir circuito em sua área de trabalho.

4.6.4. Editor de Macros

O Editor de Macros é uma ferramenta que fornece a possibilidade de construir o circuito interno de circuitos integrados que podem ser adicionados à Área de Trabalho e simulados. Os circuitos integrados criados por esse editor são chamados de portas lógicas de tipo Macros.

Para inicializar o editor deve-se selecionar o botão MACROS. Nesse momento a área de trabalho do editor é inserida na janela do programa, o Container de Portas passa a exibir o décimo primeiro grupos de ícones que contém os ícones especiais que interagem com o funcionamento do editor, e o botão MACROS passa a exibir uma imagem que indica que o editor está ativo.

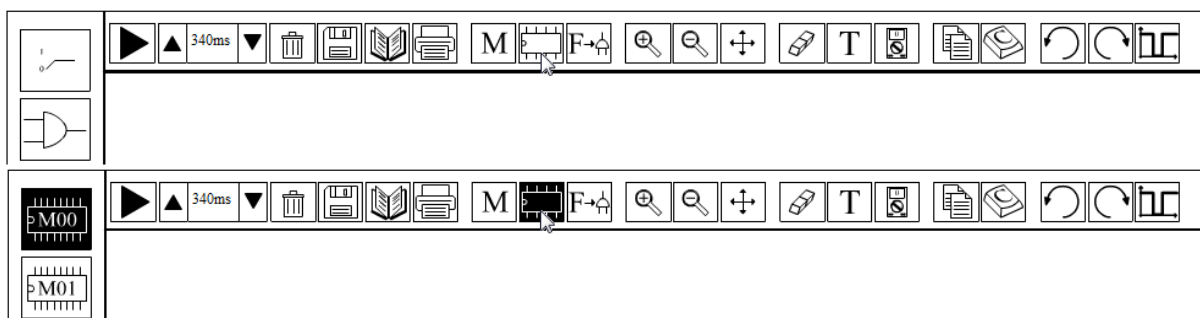


Figura 4.143 – Inicialização do Editor de Macros.

A área de trabalho do editor possui por padrão uma imagem de um circuito integrado de dezesseis pinos, e em cada um desses pinos existe uma seta de ligação que se encontra visível.

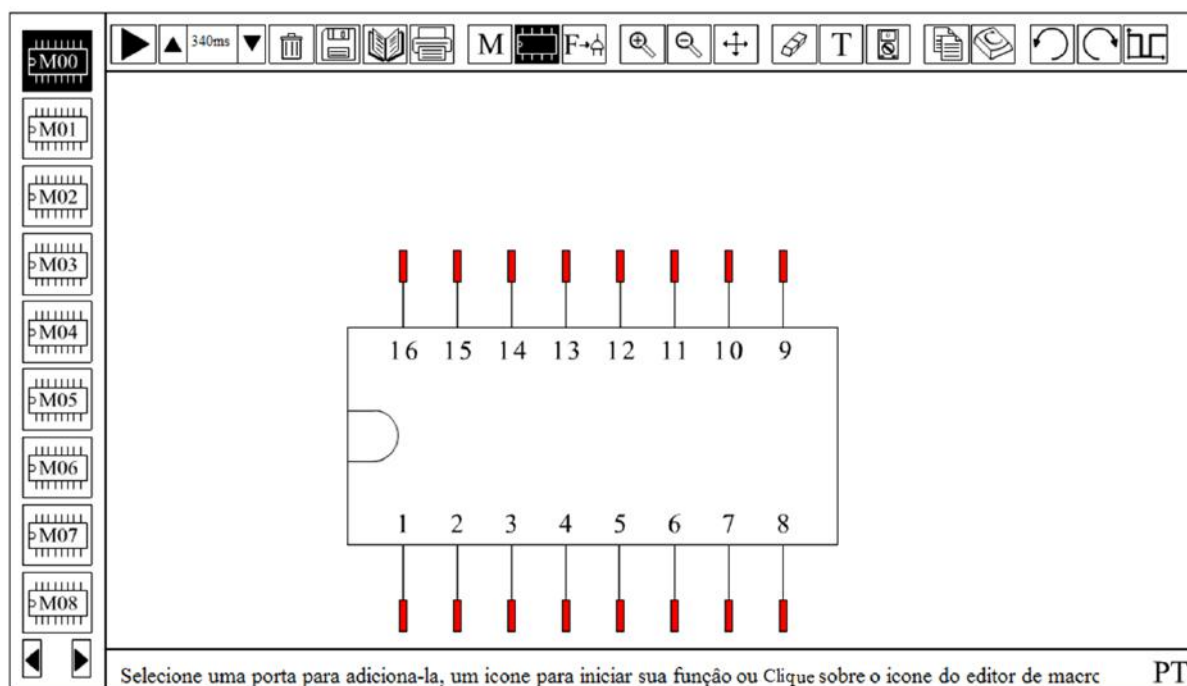


Figura 4.144 – Área de trabalho com a imagem do circuito integrado.

O editor possui nove tipos de portas lógicas de tipo Macros disponíveis para serem editadas e elas são diferenciadas por seus Números de Identificação de Macros: "M00", "M01", "M02", "M03", "M04", "M05", "M06", "M07" e "M08". Como pode ser visto na Figura 4.144, ao iniciar o editor, a porta lógica de tipo Macros "M00" é exibida por padrão e isso é indicado pelo primeiro ícone do Container de Portas que exibe a imagem com fundo de cor preta e com o seu número de identificação nela, diferente dos demais ícones que apresentam imagem original do Container de Portas. Inicialmente todas as nove portas lógicas de tipo Macros do editor encontram-se sem circuito interno.

A área de trabalho do editor interage com as ferramentas do programa da mesma maneira que a Área de Trabalho do programa, assim todas as funções do programa funcionam da mesma maneira em ambas, porém as funções Deletar, Salvar e Abrir não podem ser ativadas, não é possível executar a simulação, o Editor de Gráficos e o Editor de Atalhos não podem ser inicializados e o Editor de Minimização quando ativado só exibe a função de minimização de tabela. A criação de portas lógicas e fios na área de trabalho do editor são feitos seguindo os procedimentos descritos nos capítulos 4.5.1 e 4.5.2 respectivamente. Para acessar os ícones das portas lógicas no Container de Portas, devem-se selecionar os botões com setas para acessar o grupo de ícones que contenha o ícone da porta lógica desejada, e como foi descrito

no capítulo 4.2 os grupos de ícones que contenham ícones representativos de circuitos integrados não são exibidos e o último grupo de ícones continua a exibir os ícones especiais que possuem integração com o editor.

Após a criação do circuito interno da porta lógica de tipo Macros no editor, é necessário realizar as ligações entre os pinos das portas lógicas que compõem esse circuito e os pinos da imagem do circuito integrado, pois esse tipo de ligação é a única maneira de fazer com que os circuitos internos se comuniquem com outras portas lógicas externas ao circuito integrado. Esse procedimento é semelhante ao de criação de fios, porém algumas regras se aplicam e o resultado deste processo cria um fio especial. A primeira regra é que o fio que fará a ligação com o pino do circuito integrado só pode estar ligado a outra porta lógica, não é permitido que o fio esteja ligado a outro fio. A segunda regra é que a criação do fio deve sempre ser iniciada por um pino de uma porta lógica do circuito interno e terminar no pino do circuito integrado. Ao finalizar o fio, a seta de ligação utilizada deixa de ser exibida.

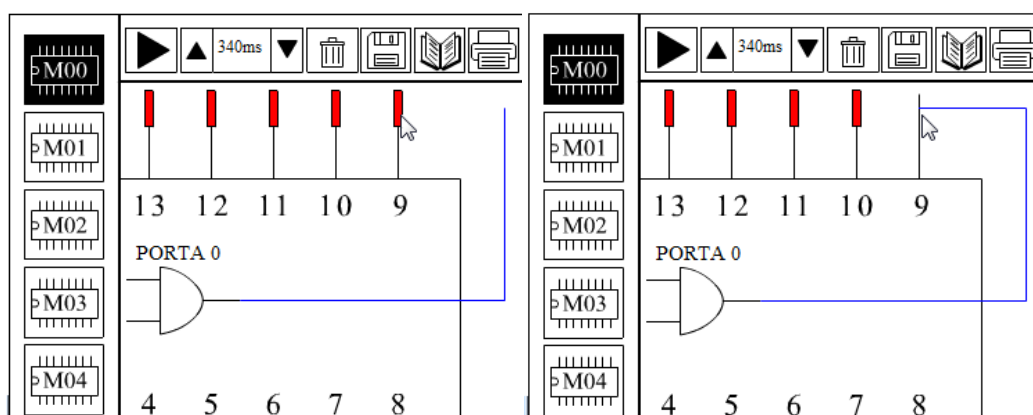


Figura 4.145 – Ligação entre o circuito integrado a o circuito interno.

O fio criado pelo processo da Figura 4.145 possui propriedades diferentes dos demais fios. Este fio não pode ser utilizado para iniciar ou finalizar uma ligação, não pode ser duplicado, não pode ser girado, mas pode ser redimensionado para acompanhar o giro e arrasto da porta lógica com o qual faz ligação. Esse fio pode ser redimensionado manualmente. Quando a porta lógica com a qual esse fio está ligado é excluída, o fio também é excluído e a Seta de Ligação do pino do circuito integrado volta a ser exibida.

Para validar as modificações feitas na área de trabalho do editor deve-se selecionar o botão MACROS, nesse momento o editor é encerrado, o botão MACROS volta a exibir sua imagem original, a área de trabalho do editor é retirada, os ícones do décimo primeiro grupo de ícones do Container de Portas voltam ao normal e os circuitos que estiverem na área de

trabalho da porta lógica de tipo Macros selecionada no editor, "M00" no caso da Figura 4.145, serão validados e incorporados às informações para esse tipo de porta lógica no editor. Assim quando o Editor de Macros for inicializado novamente, os circuitos da porta lógica de tipo Macros "M00" são exibidos na área de trabalho do editor, como estavam quando foram validados.

Os processos descritos acima foram todos feitos na porta lógica de tipo Macros "M00" do editor, porém existem nove portas lógicas de tipo Macros no editor, e esses processos podem ser feitos em todas elas. Para acessar uma porta lógica de tipo Macros diferente da que é exibida por padrão ao se inicializar o editor, deve-se posicionar o cursor do mouse sobre um dos ícones especiais do décimo primeiro grupo de ícones do Container de Portas que tenha o mesmo Número de Identificação de Macros da porta lógica que se deseja acessar, e pressionar o botão esquerdo do mouse. Nesse momento, o ícone selecionado tem sua imagem alterada para uma com fundo preto e o ícone que anteriormente possuía essa imagem passa a exibir a imagem original. Ao realizar a mudança de porta lógica de tipo Macros no editor, os circuitos da nova porta lógica passam a ser exibidos na área de trabalho do editor e é possível realizar e validar edições nesses circuitos.

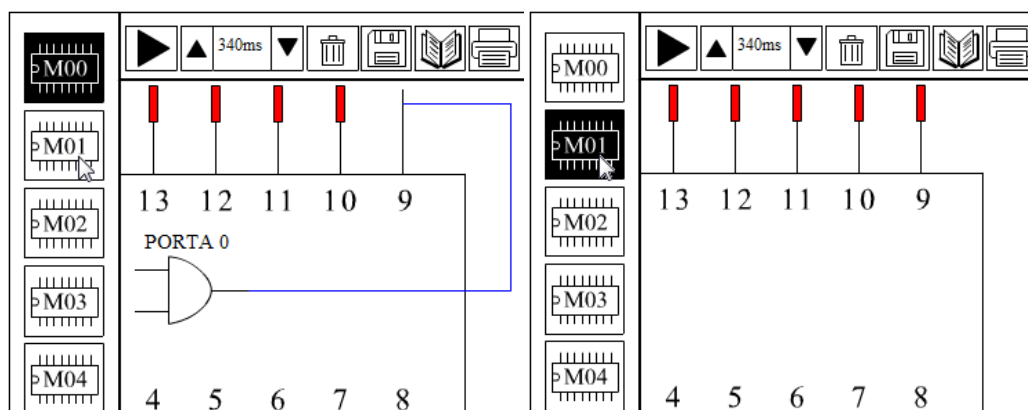


Figura 4.146 – Alteração da porta lógica de tipo Macros selecionada no editor.

Caso alguma edição tenha sido feita nos circuitos da porta lógica de tipo Macros selecionada no editor após este ter sido inicializado e essas modificações ainda não tenham sido validadas, e o processo para acessar outra porta lógica de tipo Macros no editor tenha sido realizado, todas essas modificações são perdidas. São mantidos os mesmos circuitos que essa porta lógica possuía quando houve a última validação de seus circuitos. Assim, pode-se observar que só é possível realizar a validação de uma porta lógica de tipo Macros por vez, e é necessário iniciar o editor novamente caso seja necessário editar os circuitos de outras portas lógicas de tipo Macros.

Caso o processo para acessar outra porta lógica de tipo Macros seja realizado selecionando o ícone do Container de Portas que já esteja selecionado, que apresente a imagem com fundo preto, as edições realizadas nessa porta lógica são excluídas e os circuitos exibidos passam a ser os mesmo que essa porta lógica possuía quando houve a última validação sobre ela, assim esse processo pode ser utilizado como uma maneira de reiniciar a porta lógica de tipo Macros do editor para as suas configurações da última validação.

É possível realizar a duplicação de um circuito da Área de Trabalho do programa para a área de trabalho do Editor de Macros. Para realizar essa duplicação é necessário que, no momento em que se inicializa o editor, o Grupo Selecionado já contenha todas as portas lógicas e fios a serem duplicados, assim a informação do Grupo Selecionado ficará armazenada e toda vez que a função de duplicação for realizada esse grupo será duplicado. Caso alguma porta lógica ou fio da área de trabalho do editor estejam selecionados quando a função de duplicação for realizada, a informação do Grupo Selecionado na Área de Trabalho do programa antes da inicialização do editor, é perdida e o Grupo Selecionado na área de trabalho do editor é duplicado.

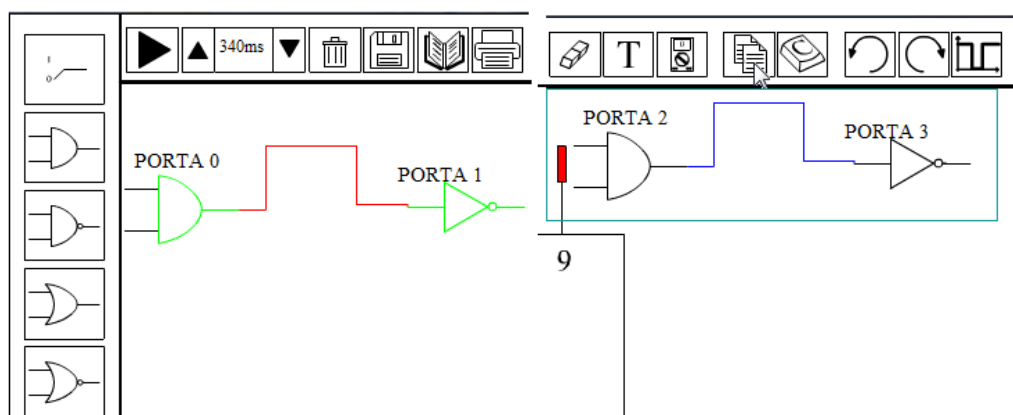


Figura 4.147 – Duplicação no Editor de Macros, do Grupo Selecionado da Área de Trabalho do programa.

Caso haja no grupo duplicado algum circuito integrado ou porta lógica de tipo Macros, o processo de duplicação desse grupo é cancelado, uma vez que esses tipos de portas lógicas não podem ser inseridos na área de trabalho do editor. Chaves lógicas, portas lógicas de tipo Led e portas lógicas de tipo Pulser podem ser inseridas na área de trabalho do editor, porém quando o processo de validação do circuito é realizado, o editor informa que não pode ser encerrado enquanto portas lógicas desses tipos existirem em sua área de trabalho.

Com o editor encerrado, o décimo primeiro grupo de ícones do Container de Portas volta a exibir os ícones originais, e estes representam os nove tipos de portas lógicas de tipo Macros do Editor de Macros. Caso algum desses ícones seja utilizado no processo de criação de portas lógicas, uma porta lógica de tipo Macros "MXX" será criada, onde "XX" é substituído pelo Número de Identificação de Macros do ícone que foi utilizado para criar a porta lógica. O circuito interno dessa porta lógica será o mesmo circuito presente na porta lógica de tipo Macros do Editor de Macros que possui o mesmo Número de Identificação de Macros da porta lógica criada.

Como foi descrito no capítulo 4.6.1, o Editor de Propriedades de uma porta lógica de tipo Macros permite a visualização de seus circuitos internos. Assim é possível checar que a porta lógica criada realmente mantém os mesmos circuitos internos criado no Editor de Macros. No capítulo 4.6.1 também foi descrita a presença de um marcador que está ligado à opção de sincronização entre a porta lógica e o Editor de Macros. Essa sincronização se refere à atualização dos circuitos internos de uma porta lógica de tipo Macros que tenha esse marcador ativo, para que seus circuitos internos sejam substituídos pelos novos circuitos validados da porta lógica de tipo Macros do Editor de Macros de mesmo Número de Identificação de Macros da porta lógica.

Para que a sincronização das portas lógicas de tipo Macros que possuem o marcador de sincronização ativado seja realizada é necessário que um segundo marcador esteja ativo. O segundo marcador de sincronização encontra-se na ranhura da imagem do circuito integrado na área de trabalho do Editor de Macros, e para ativá-la deve-se posicionar o cursor do mouse sobre o marcador e pressionar o botão esquerdo do mouse. Nesse momento o marcador passa a exibir um fundo preto.

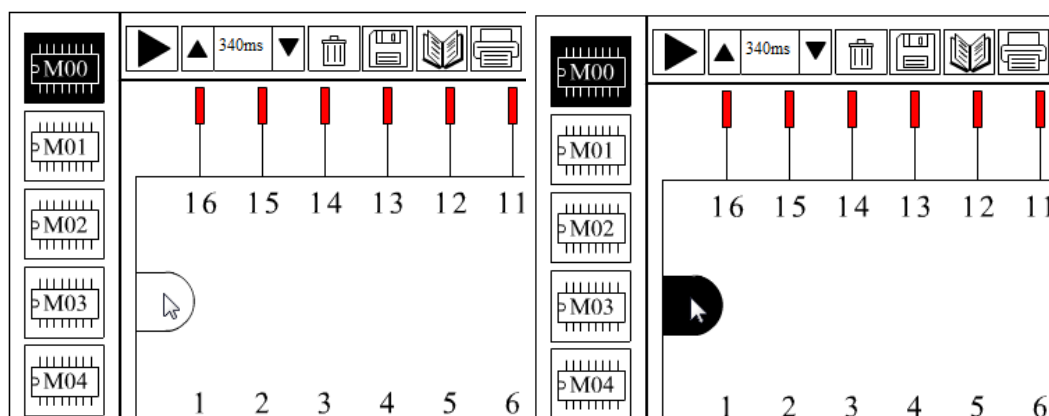


Figura 4.148 – Ativação do segundo marcador de sincronização.

Diferente do marcador presente no Editor de Propriedades, o segundo marcador não se mantém ativado após o encerramento do Editor de Macros, sendo necessário reativá-lo todas as vezes que o processo de sincronização seja realizado. Com o marcador da ranhura da imagem do circuito integrado ativo, após a validação dos circuitos de uma porta lógica de tipo Macros do Editor de Macros, todas as portas lógicas de tipo Macros na Área de Trabalho que tenham o mesmo Número de Identificação de Macros e que tenham o marcador de sincronização ativado, terão seus circuitos internos substituídos por circuitos iguais ao que foi validado. Porém, antes dessa substituição ocorrer, o editor informa ao usuário quais portas lógicas terão seus circuitos internos modificados e permite que esse processo seja cancelado ou confirmado.

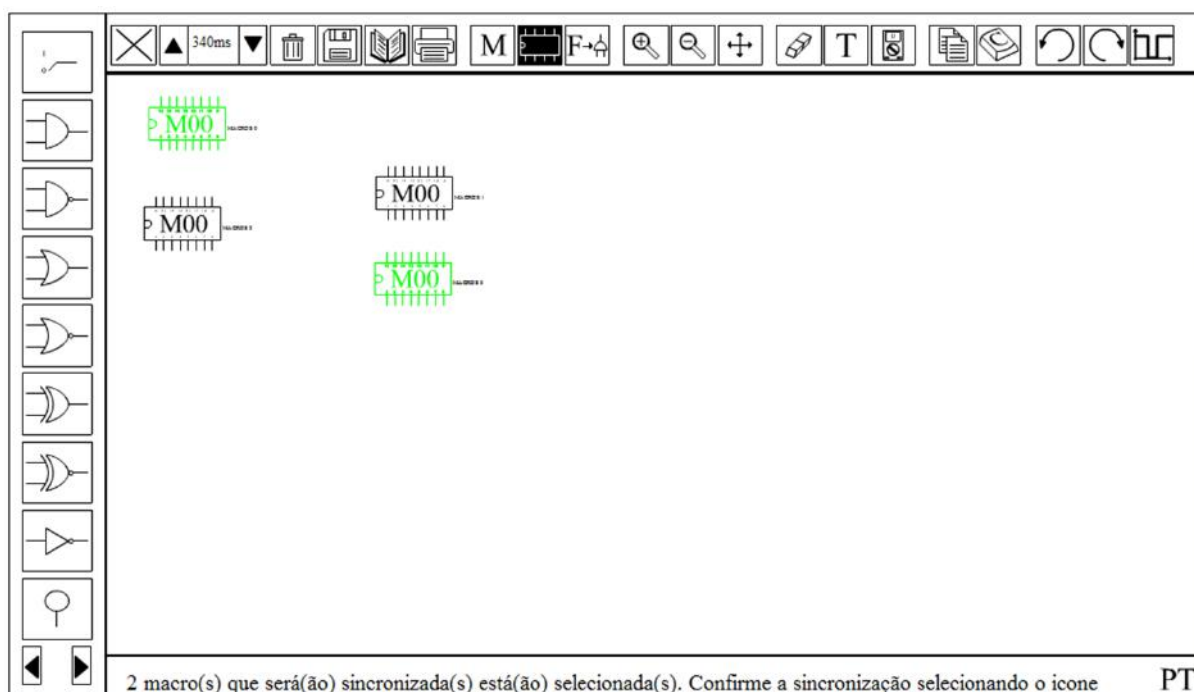


Figura 4.149 – Confirmação de sincronização.

As portas lógicas que serão sincronizadas são selecionadas e sua cor é mudada para amarelo, a Área de Trabalho é reduzida para suas dimensões mínimas e somente as funções Arrasto, Ampliar e Reduzir são liberadas. Para cancelar a sincronização deve-se selecionar o botão INICIAR/CANCELAR que exibe a imagem "X", e para confirmar o processo deve-se selecionar o botão MACROS. Assim as portas lógicas selecionadas voltam a exibir sua cor original e os botões MACROS e INICIAR/CANCELAR voltam a exibir suas imagens originais. No caso de cancelamento a validação das modificações no Editor de Macros é realizada, porém as portas lógicas de tipo Macros que seriam sincronizadas mantêm seus circuitos internos inalterados. No caso de confirmação da sincronização, além da validação

das edições no Editor de Macros, os circuitos das portas lógicas selecionadas são substituídos pelo circuito validado.

Todas as vezes que houver o processo de sincronização de uma porta lógica de tipo Macros do Editor de Macros, todas as portas lógicas que tiverem o mesmo Número de Identificação de Macros e que tiverem os marcadores de sincronização ativos terão seus circuitos substituídos. Assim se o processo de sincronização for feito para a primeira porta lógica de tipo Macros do Editor de Macros, somente as portas lógicas que forem de tipo "M00" serão sincronizadas. Caso não existam portas lógicas a serem sincronizadas após o processo de sincronização ser executado, este é automaticamente cancelado.

4.6.5. Editor de Gráficos

Através do Editor de Gráficos é possível monitorar o estado lógico dos pinos das portas lógicas na Área de Trabalho. O monitoramento pode ser feito em tempo real durante a execução da simulação ou após esta ter sido encerrada, através da análise dos gráficos que são criados na janela do editor.

Para inicializar o editor, supondo que a simulação não esteja em execução, deve-se selecionar o botão GRÁFICOS e nesse momento a janela do editor é inserida na Área de Trabalho abaixo da Barra de Ferramentas e o botão GRÁFICOS passa a exibir uma imagem que informa que o editor está ativo.

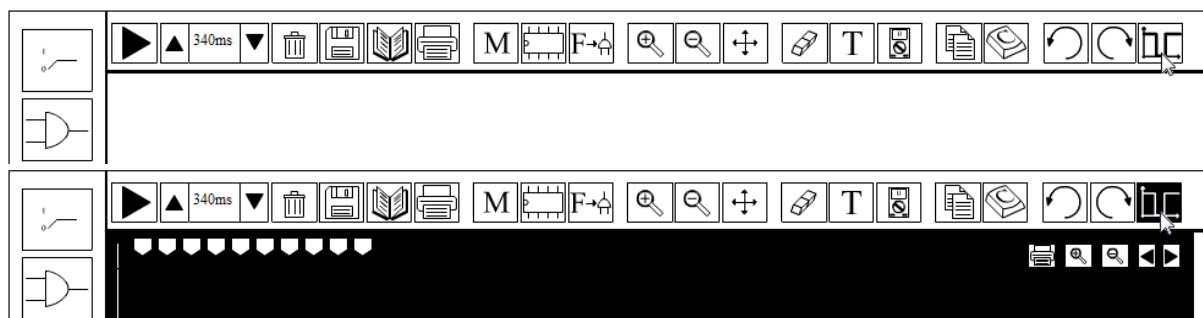


Figura 4.150 – Inicialização do Editor de Gráficos.

A janela do editor exibe um par de eixos orientados onde os gráficos são criados, um conjunto de dez bandeiras localizadas na parte superior esquerda da janela, um conjunto de botões de navegação localizados na parte superior direita da janela, e um botão para impressão dos gráficos localizado próximo aos botões de navegação. É possível observar na

Figura 4.151 que a janela do editor possui dimensões reduzidas, ocupando somente uma pequena porção da janela do programa. Essa característica faz com que seja possível visualizar a Área de Trabalho e os circuitos nela contidos, uma vez que isso é necessário para a execução de algumas funções do editor.

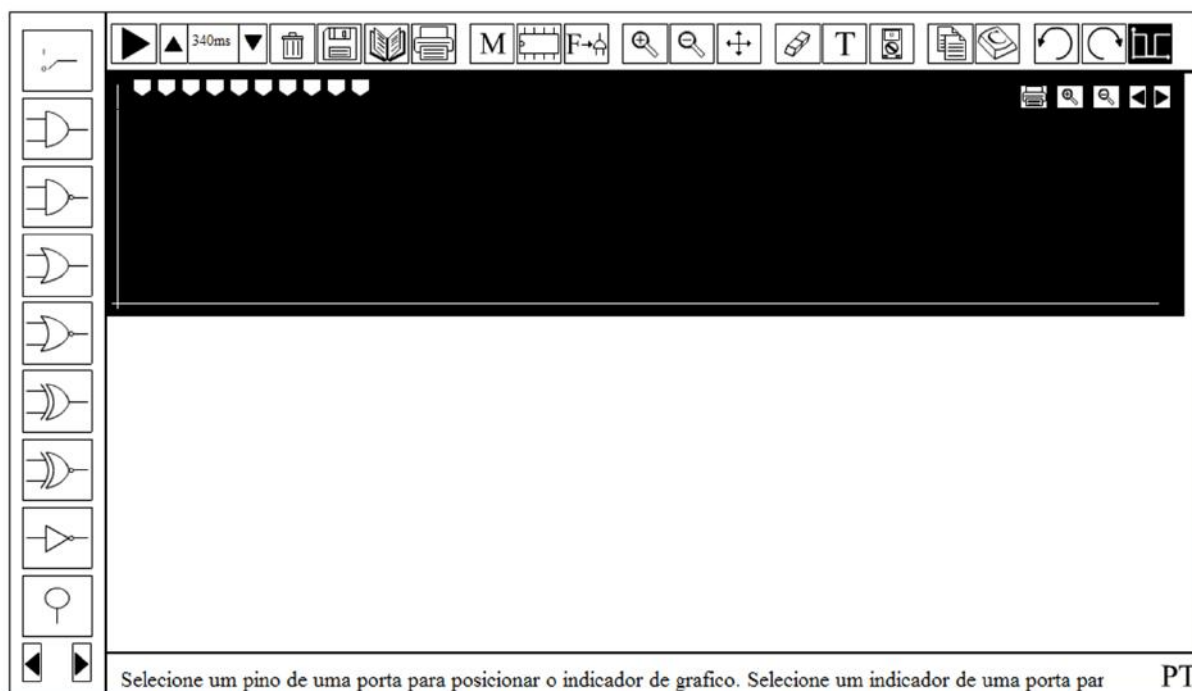


Figura 4.151 – Janela do Editor de Gráficos.

Ao iniciar o editor todas as dez bandeiras encontram-se com a cor branca, como pode ser observado na Figura 4.151. Isso indica que nenhum pino de porta lógica está sendo monitorado, e assim nenhum gráfico será criado no eixo orientado quando a simulação dos circuitos for executada. Para que um pino seja monitorado ele deve ser selecionado, com o editor de gráficos inicializado e a simulação fora de execução, posicionando o cursor do mouse sobre ele, pressionando e soltando o botão esquerdo do mouse. Isso fará com que uma Bandeira de Monitoramento seja posicionada junto a esse pino, e também resultará na mudança da primeira bandeira da janela do editor, que passará a exibir a mesma imagem da bandeira que se encontra junto ao pino da porta lógica.

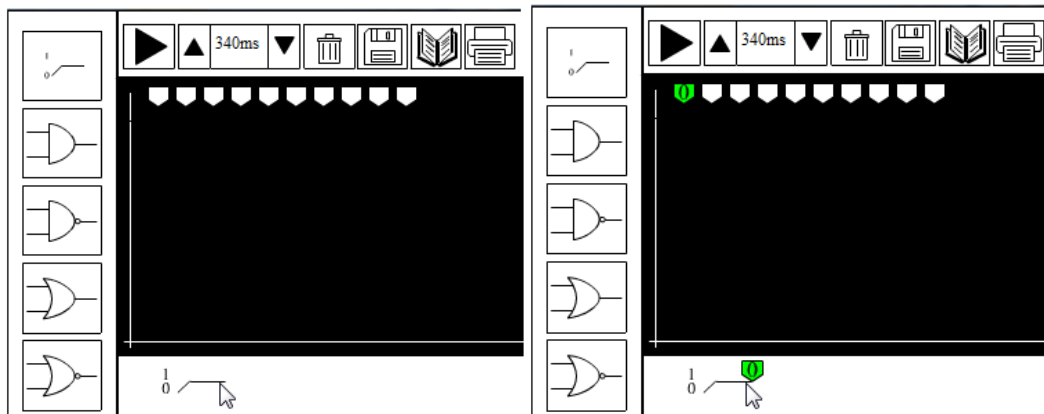


Figura 4.152 – Inserção da Bandeira de Monitoramento em um pino.

As duas bandeiras indicam que o primeiro gráfico, de número de identificação zero, será construído com base nas informações dos estados lógicos que o pino monitorado apresentar durante a execução da simulação. As bandeiras possuem fundo de cor verde, e essa é a cor com que o gráfico será construído na janela do editor.

Caso o processo de inserção de Bandeiras de Monitoramento seja repetido para outros pinos, as demais bandeiras do editor serão posicionadas junto a esses pinos, e as bandeiras da janela do editor serão substituídas. A Figura 4.153 mostra uma situação em que todas as bandeiras foram inseridas e é possível observar que cada bandeira possui uma cor e um número diferentes, e estes refletem a cor com a qual cada gráfico será construído na janela do editor de gráficos. As bandeiras são inseridas em ordem crescente de número de identificação.

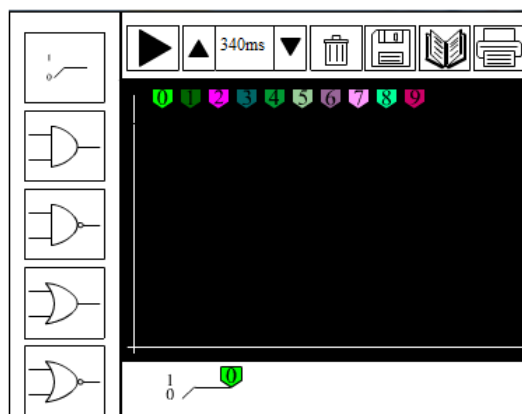


Figura 4.153 – Todas as bandeiras de monitoramento inseridas.

Caso seja necessário retirar uma bandeira de monitoramento de um pino, deve-se posicionar o cursor do mouse sobre a bandeira, pressionar e soltar o botão esquerdo do mouse. Assim a bandeira selecionada é retirada da Área de Trabalho, e a bandeira que possui o

mesmo número de identificação na janela do editor, volta a exibir a imagem da bandeira branca. Assim o gráfico referente à bandeira retirada não é construído na janela do editor.

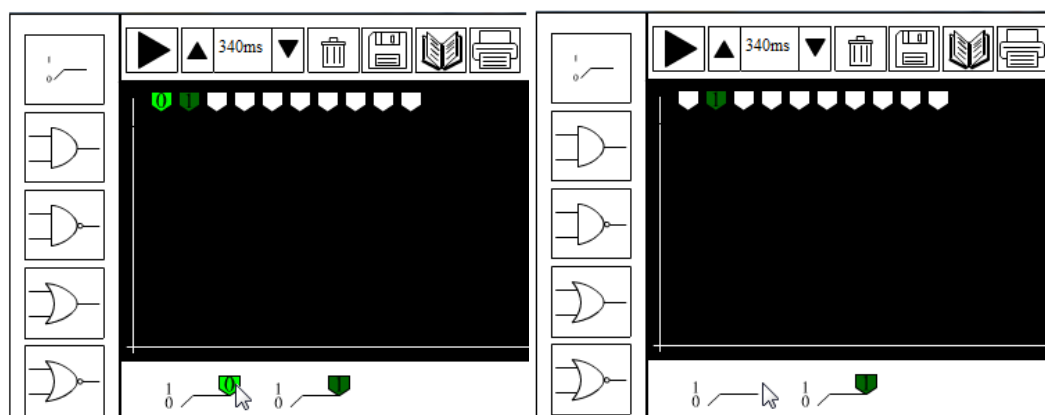


Figura 4.154 – Retirada de uma bandeira de monitoramento.

Na Figura 4.154, somente a bandeira de número de identificação igual a um está inserida na Área de Trabalho, assim se o processo de inserção de Bandeiras de Monitoramento for feito nessa situação, as bandeiras serão inseridas em ordem crescente, levando-se em consideração somente as bandeiras que ainda não estejam inseridas na Área de Trabalho. Assim a próxima bandeira a ser inserida seria a bandeira de número de identificação igual a zero, seguida da bandeira de número de identificação igual a dois.

Enquanto a janela do editor estiver inserida na Área de Trabalho, é possível utilizar as função de arrasto, ampliação e redução na Área de Trabalho para poder alcançar uma região desta que não é exibida. Mesmo que essas funções sejam aplicadas, a janela do editor permanece em sua posição original e mantém as mesmas dimensões. Porém existe uma maneira de mudar a posição da janela do editor, fazendo com que a janela passe a ter sua base alinhada com a Barra de Informações. Para realizar esse reposicionamento deve-se posicionar o cursor do mouse na região da janela do editor abaixo das bandeiras e dos botões, pressionar e soltar o botão esquerdo mouse.

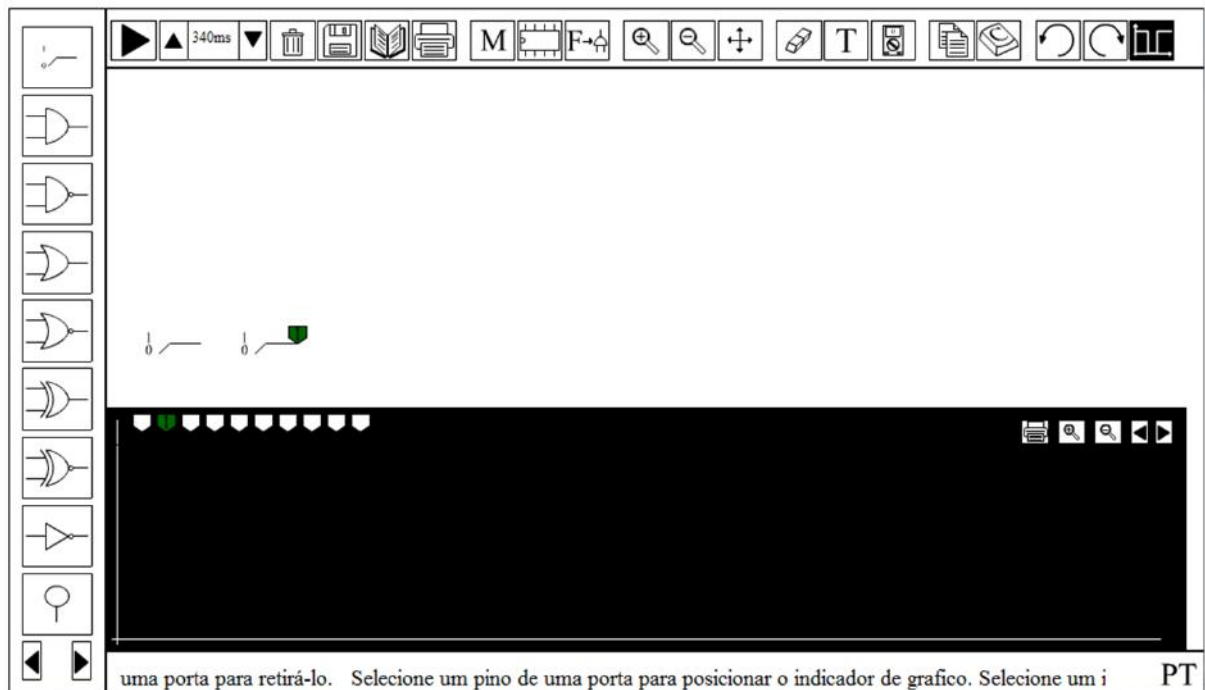


Figura 4.155 – Janela do Editor de Gráficos reposicionada.

Além das funções de arrasto, ampliação e redução da Área de Trabalho, apenas a execução da simulação pode ser realizada enquanto o editor estiver ativo. Caso a execução da simulação seja iniciada, serão construídos, no eixo orientado da janela do editor, os gráficos referentes as bandeiras que estiverem inseridas na Área de Trabalho. À medida que as iterações da simulação forem realizadas, os gráficos são construídos com base no estado lógico do pino monitorado pela bandeira referente a esse gráfico. O eixo vertical representa o estado lógico, onde o estado lógico zero encontra-se na interseção desse eixo com o eixo horizontal e o estado um encontra-se um pouco baixo das bandeiras. O eixo horizontal representa o tempo, e nesse eixo são exibidas marcações com informações sobre o tempo em segundos.

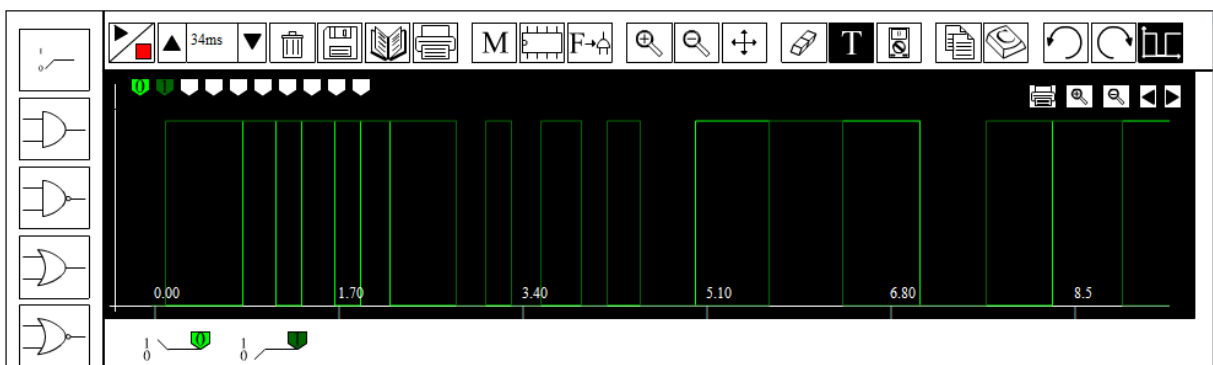


Figura 4.156 – Gráficos construídos na janela do Editor de Gráficos.

Na Figura 4.156 é possível observar que os gráficos são sobrepostos na janela do editor, pois o espaço reduzido da janela impede que gráficos sejam exibidos separadamente. Porém é possível selecionar quais gráficos devem ser exibidos na janela do editor de gráficos. Essa seleção é feita com base nas bandeiras da janela. É possível desativar a exibição do gráfico referente a uma bandeira sem a necessidade de retirar a bandeira de monitoramento referente a esta da Área de Trabalho. Nesse caso o gráfico referente a essa bandeira continua a ser construído, porém não é exibido na janela do editor. Para realizar esse processo deve-se posicionar o cursor do mouse sobre uma das bandeiras numerada exibidas na janela do editor, pressionar e soltar o botão esquerdo do mouse. Assim essa bandeira é retirada e a bandeira branca é exibida em seu lugar, e nesse momento o gráfico referente a essa bandeira deixa de ser exibido na janela do editor de gráficos.

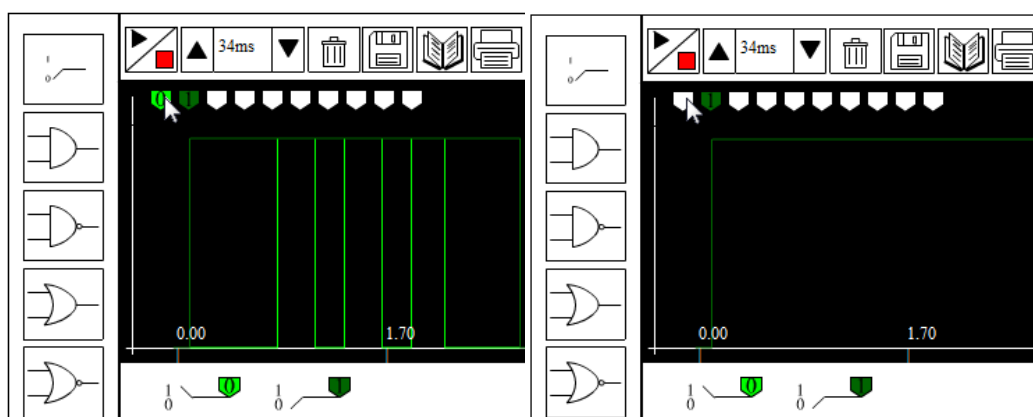


Figura 4.157 – Desativação de um gráfico.

O processo realizado na Figura 4.157 pode ser repetido quantas vezes sejam necessárias, mesmo que resulte na desativação de todos os gráficos disponíveis. Porém esse processo só pode ser aplicado sobre uma bandeira que tenha uma Bandeira de Monitoramento de mesmo número de identificação inserida na Área de Trabalho.

Para reativar a exibição de um gráfico deve-se realizar o mesmo procedimento observado na Figura 4.157, porém selecionando uma bandeira que teve a exibição de seu gráfico desativada. Essa bandeira será substituída pela bandeira numerada e o gráfico volta ser exibido na janela do editor de gráficos.

A janela do editor de gráficos possui um modo expandido onde os gráficos exibidos não são sobrepostos. Para ativar esse modo deve-se posicionar o cursor do mouse na região da janela do editor abaixo das bandeiras e dos botões, pressionar e soltar o botão direito do mouse. Assim a janela aumenta sua dimensão vertical e passa a exibir os gráficos dispostos

verticalmente. Nesse modo a janela do editor de gráficos não pode ter sua posição modificada, ela permanece junto à Barra de Ferramentas.

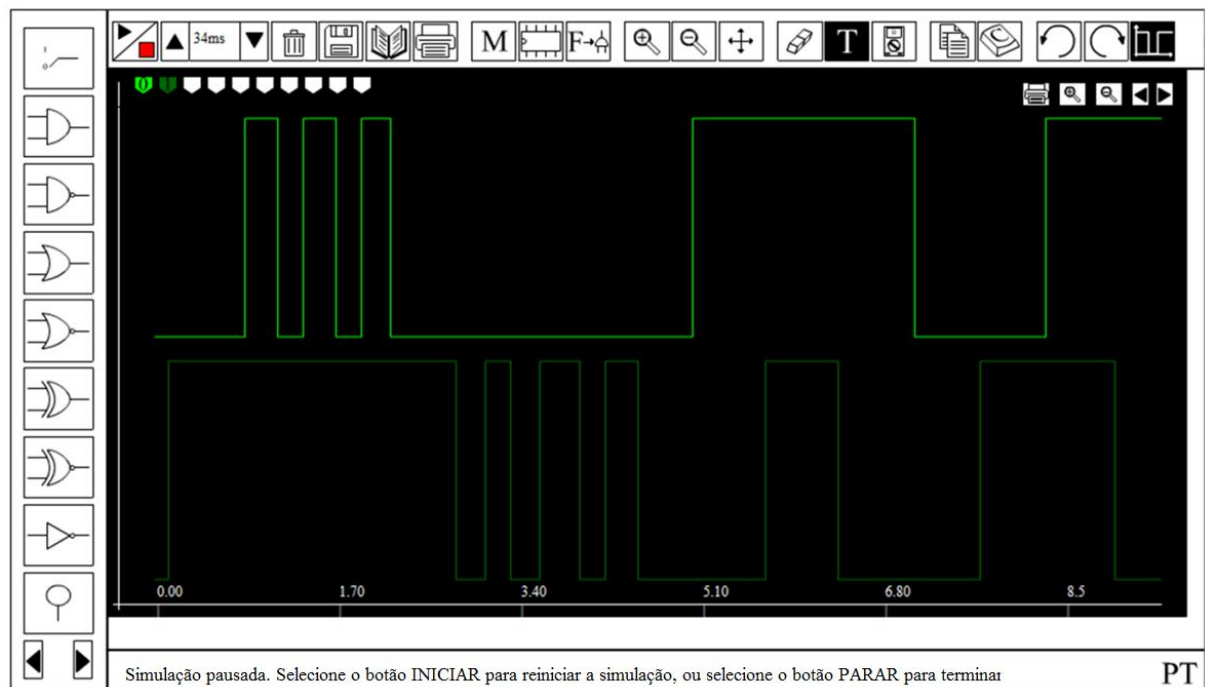


Figura 4.158 – Modo expandido da janela do Editor de Gráficos.

Os gráficos da Figura 4.158 são dispostos verticalmente e são redimensionados para que todos caibam na janela. Estes gráficos são dispostos em ordem crescente de número de identificação, onde o gráfico mais acima na janela é sempre o gráfico com menor número de identificação que não esteja com a exibição desativada. Somente os gráficos que não estão com a exibição desativada são dispostos na janela do editor de gráficos.

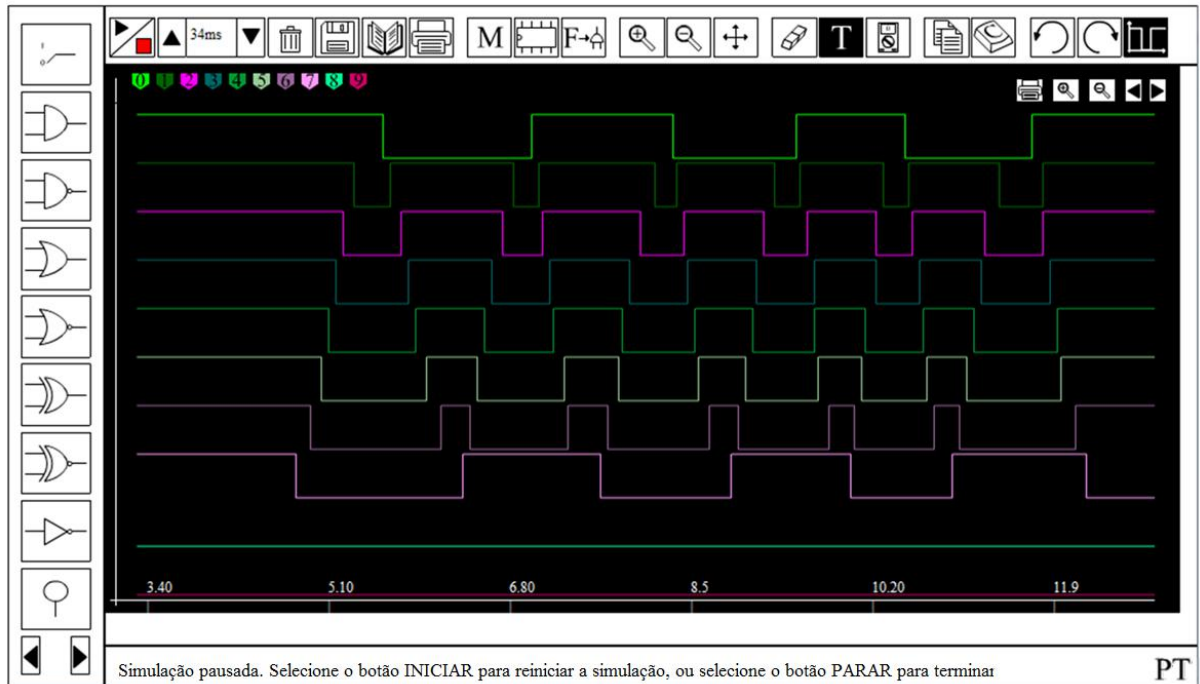


Figura 4.159 – Todos os gráficos exibidos na janela do Editor de Gráficos em modo estendido.

Caso o processo de desativação da exibição de um gráfico seja realizado, esse gráfico deixa de ser exibido, e os demais são redimensionados para ocupar toda a área da janela do editor de gráficos. A Figura 4.160 ilustra a janela do editor da Figura 4.159 com as bandeiras de número de identificação 3, 4, 5 e 6 desativadas.

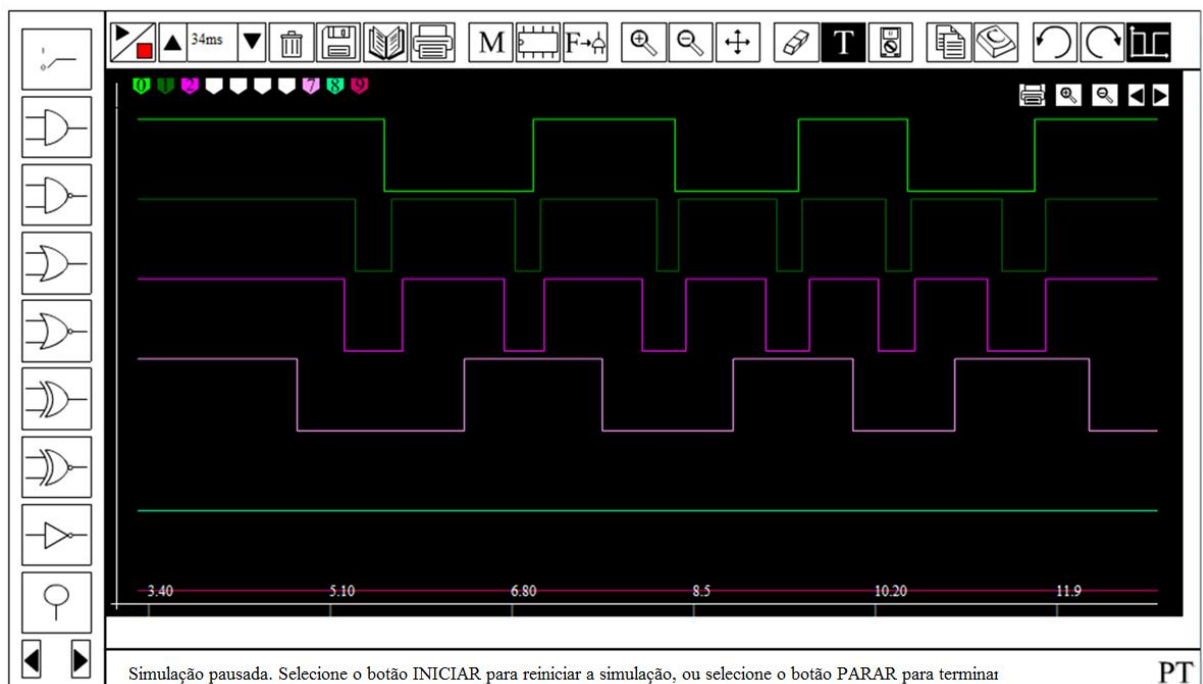


Figura 4.160 – Desativação de gráficos no modo estendido da janela do editor de gráficos.

Caso a exibição de um gráfico seja reativada ele volta a ser exibido ocupando a posição que seu número de identificação indica e os demais gráficos são redimensionados para que todos caibam na janela do editor de gráficos.

Os gráficos criados permanecem na janela do editor mesmo que a execução da simulação seja encerrada, e só são apagados quando uma nova simulação é iniciada, ou quando a Bandeira de Monitoramento referente a um desses gráficos é retirada da Área de Trabalho. Os botões de navegação da janela do editor de gráficos só podem ser utilizados quando a simulação estiver pausada ou quando a simulação não estiver em execução, e permitem analisar as informações exibidas no editor de gráficos. Suas funções permitem aumentar e diminuir a proporção com que o eixo horizontal é exibido, através do botão que exibe a imagem de uma lupa com o sinal de adição e do botão que exibe a imagem de uma lupa com o sinal de subtração, respectivamente.

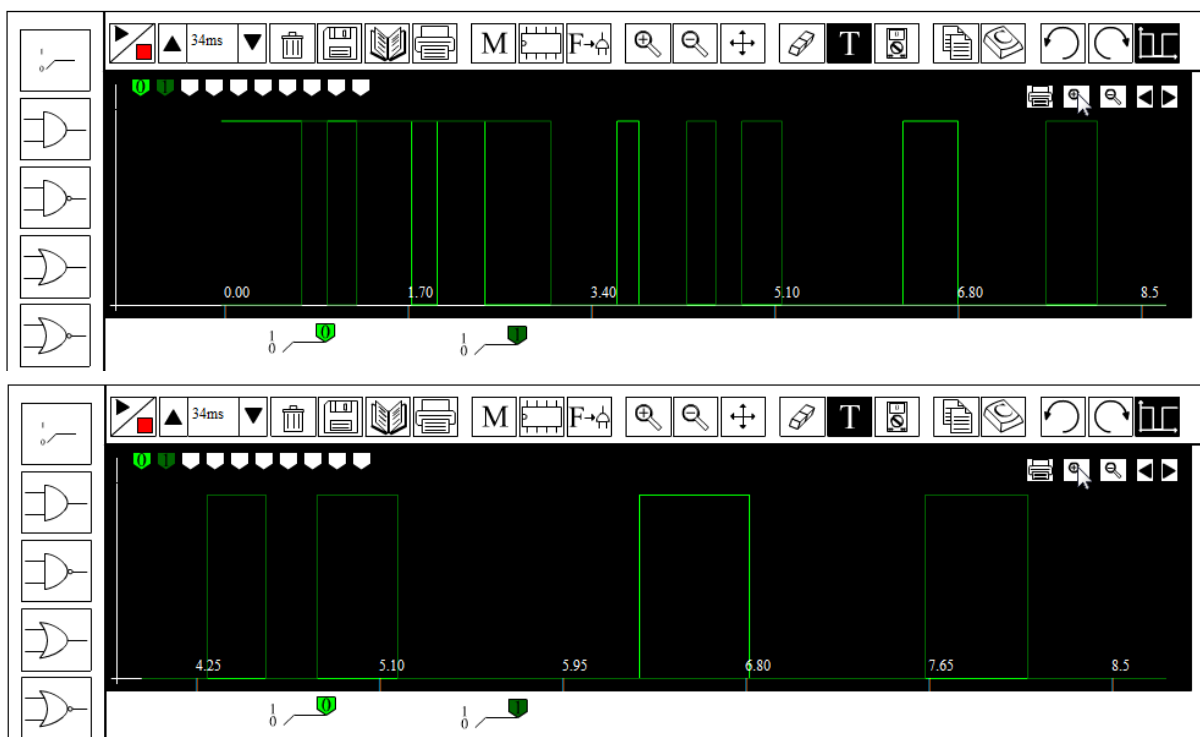


Figura 4.161 – Mudança na proporção do eixo horizontal.

Os botões que exibem imagens de setas direcionadas para a direita e esquerda, são responsáveis por excursionar por toda a extensão dos gráficos avançando e retornando o tempo exibido no eixo horizontal.

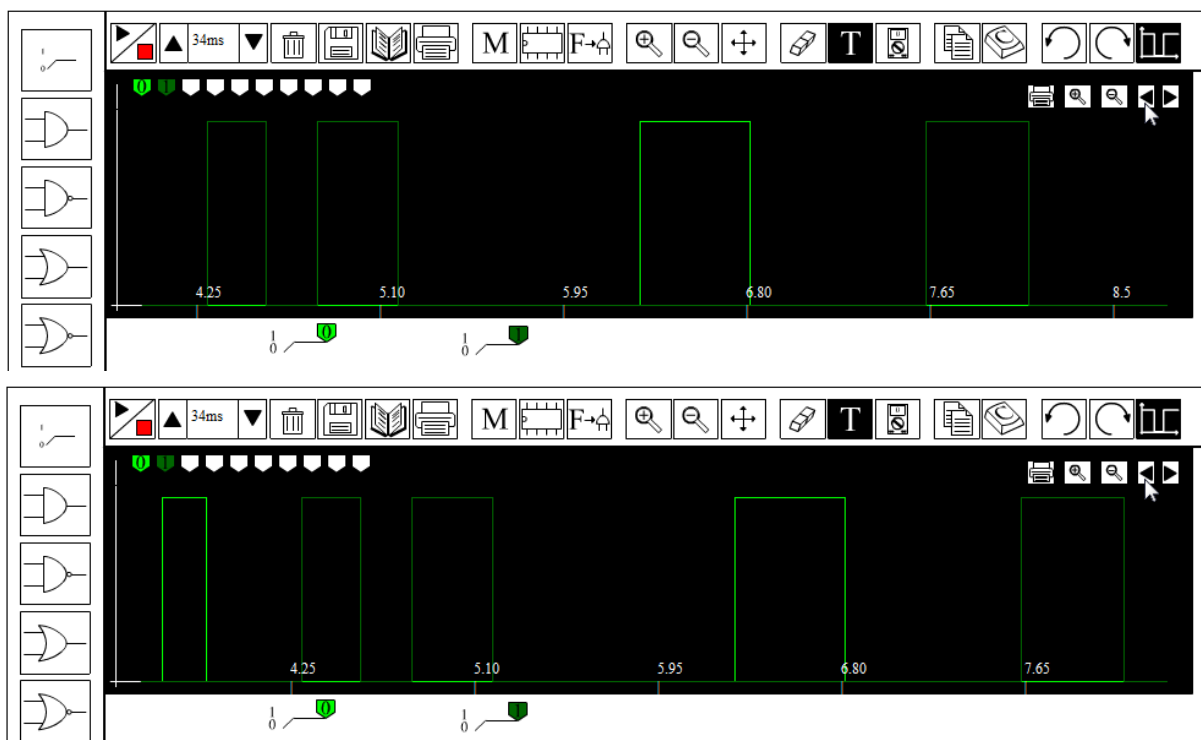


Figura 4.162 – Avanço do tempo exibido no eixo horizontal.

O botão de impressão permite realizar a impressão dos gráficos da janela do editor. Quando este botão é selecionado, a janela de impressão do editor é inserida na Área de Trabalho, e nela deve-se informar qual trecho do gráfico deve ser impresso. A janela apresenta duas caixas de texto, a primeira próxima ao texto "Iteração Inicial" e a segunda próxima ao texto "Iteração final".

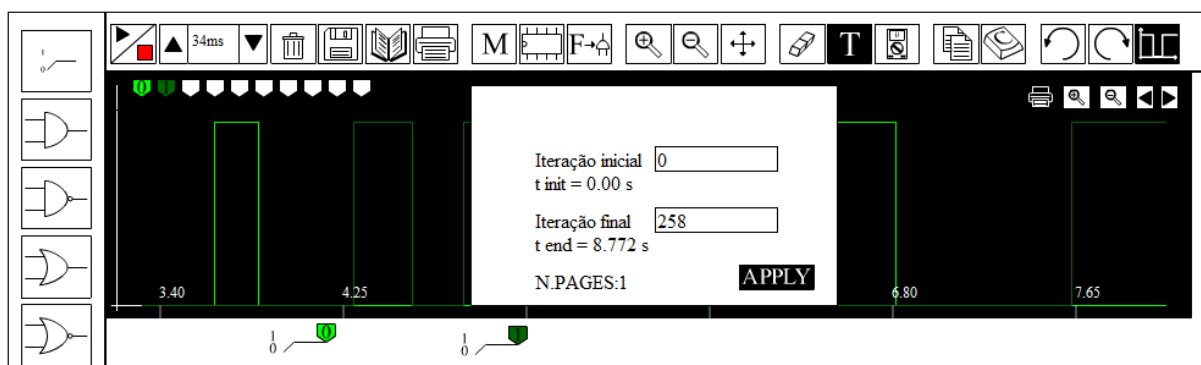


Figura 4.163 – Janela de impressao do editor de gráficos.

Nessas caixas de texto deve-se informar o número das iterações que delimitam o trecho do gráfico a ser impresso. Assim que o número da iteração é inserido em uma caixa de texto, inicialmente é realizada uma checagem para identificar incoerências como o número da iteração final ser menor do que o numero da iteração inicial e número de iteração não existente. Caso qualquer incoerência seja encontrada os números nas caixas são editados

automaticamente para valores que corrijam as incoerências. Em seguida, os textos abaixo das caixas de texto informam em que tempo a iteração informada foi realizada durante a simulação e na base da janela é exibida a quantidade de páginas que serão necessárias para realizar a impressão.

É necessário que haja algum gráfico construído para que a impressão possa ser realizada e somente os gráficos que não tenham sua exibição desativada serão impressos. A versão dos gráficos que será impressa, é aquela que se encontra no modo expandido da janela do editor de gráficos, onde os gráficos são dispostos verticalmente e a proporção em que o eixo horizontal será impresso é a mesma que é exibida na janela do editor no momento da confirmação da impressão. Assim quanto maior for essa proporção mais folhas serão necessárias para imprimir um mesmo trecho do gráfico.

Para confirmar a impressão deve-se posicionar o cursor do mouse sobre o botão APPLY da janela de impressão, pressionar e soltar o botão esquerdo do mouse. Assim a janela de impressão do sistema operacional é chamada e o programa fica inacessível até que esta janela seja encerrada, quando a janela de impressão do editor é retirada da Área de Trabalho e o botão de impressão da janela do editor de gráficos volta a exibir sua imagem original. Caso a impressão seja confirmada na janela de impressão do sistema operacional, são impressas todas as paginas necessárias para cobrir o trecho do gráfico informado. Uma página impressa pelo editor está no Apêndice A.

Para cancelar a impressão deve-se selecionar o botão GRÁFICOS o que faz com que a janela de impressão seja retirada da Área de Trabalho e o botão de impressão da janela do editor de gráficos volte a exibir sua imagem original.

Na página do Apêndice A é possível observar que cada página é dividida em três raias dispostas verticalmente. A raia mais acima na folha contém as primeiras iterações, tempo de simulação menor, a raia do meio segue com as iterações seguintes, e a raia mais abaixo exibe as iterações finais da página, que correspondem a valores de tempo mais elevados que as demais raias. As páginas seguintes mantêm o mesmo padrão de exibição de raias, porém iniciando da iteração final da folha antecedente. Em cada raia as bandeiras dos gráficos são exibidas no canto esquerdo da página, e abaixo dos gráficos de cada raia existe um eixo orientado mostrando a divisão do tempo na folha.

Para se encerrar o Editor de Gráficos deve-se proceder assim como foi feito para encerrar a janela de impressão do editor de gráficos, selecionar o botão GRÁFICOS que apresenta a imagem de fundo preto, porém nesse caso a janela do editor de gráficos, será retirada da Área de Trabalho. O processo de encerramento do Editor de Gráficos pode ser realizado com a simulação sendo executada ou não. Mesmo após o encerramento do editor, as Bandeiras de Monitoramento continuam inseridas na Área de Trabalho, e quando uma nova simulação é executada os gráficos referentes a essas bandeiras são construídos no editor. Assim quando o editor de gráficos for inicializado novamente os gráficos da última simulação estarão em sua janela.

4.6.6. Editor de Impressão

O Editor de Impressão é uma ferramenta que permite imprimir os circuitos da Área de Trabalho do programa, os circuitos da área de trabalho do Editor de Macros e os circuitos do interior de uma porta lógica de tipo Macros. Em todas essas situações é possível escolher que trecho do circuito deve ser impresso.

Para se iniciar o editor no caso da impressão da Área de Trabalho do programa deve-se selecionar o botão IMPRIMIR, com nenhuma outra função ou editor ativo. No caso da impressão da área de trabalho do Editor de Macros, o processo descrito deve ser repetido, porém com o Editor de Macros inicializado. No caso dos circuitos internos de uma porta lógica de tipo Macros, deve-se recorrer ao Editor de Propriedades, como foi descrito no capítulo 4.6.1.

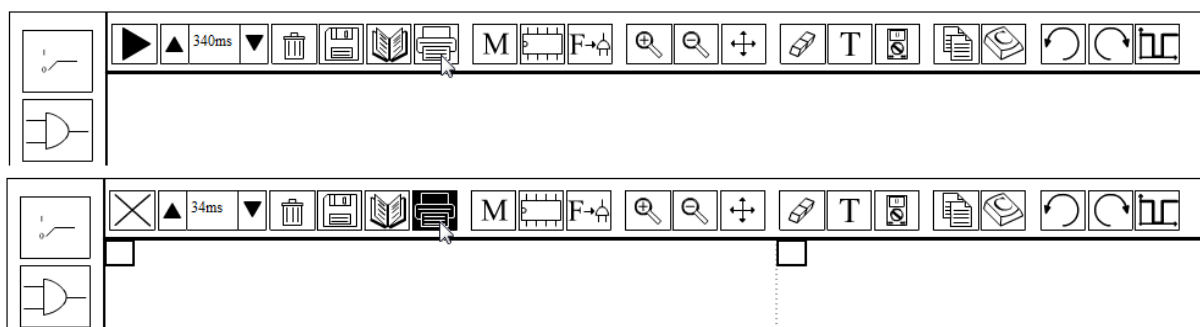


Figura 4.164 – Inicialização do Editor de Impressão.

Após a ativação do editor o botão IMPRIMIR passa a exibir uma imagem que indica que esse editor está ativo, o botão INICIAR/CANCELAR passa a exibir a imagem de um "X", a Área de Trabalho é redimensionada para suas menores dimensões e somente a função de arrasto da Área de Trabalho pode ser utilizada para que ela possa ser visualizada totalmente.

O Editor de Impressão não possui uma janela como os demais editores, ele interage com as áreas de trabalho. Essa interação ocorre por meio da divisão dessas áreas de trabalho em regiões que representam as folhas que serão impressas. Cada região é delimitada por linhas pontilhadas que representam as dimensões de uma folha de tipo A4, e em cada uma delas é inserido um marcador posicionado no seu canto superior esquerdo. Existem doze regiões que cobrem toda a Área de Trabalho e a função de arrasto deve ser utilizada para que o marcador de cada região possa ser exibido na janela do programa. Deve-se posicionar o cursor do mouse sobre o marcador da região que se deseja imprimir e pressionar o botão esquerdo do mouse, o que faz com que o marcador seja ativado, mudando sua imagem de fundo branco por uma de fundo preto para indicar que se encontra ativo.

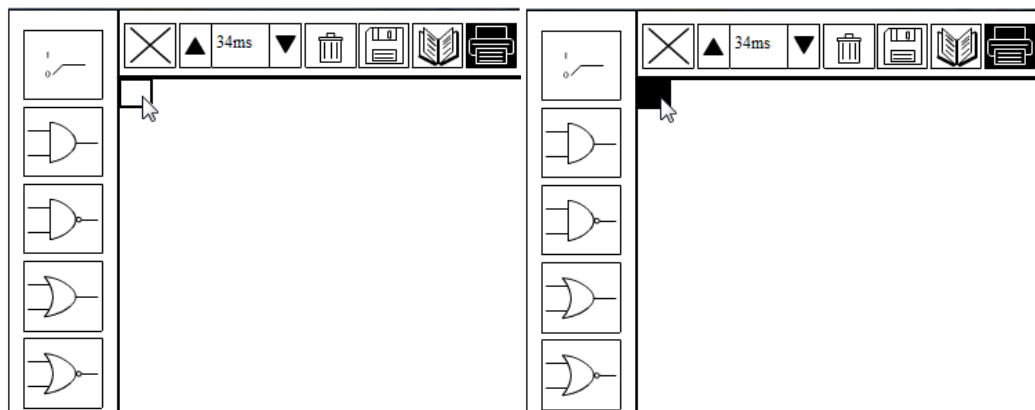


Figura 4.165 – Ativação do marcador de impressão.

Devem-se ativar os marcadores das regiões que se deseja imprimir e em seguida confirmar o processo de impressão selecionando o botão IMPRIMIR e nesse momento a janela de impressão do sistema operacional é chamada e o programa fica inacessível até que essa janela seja encerrada.

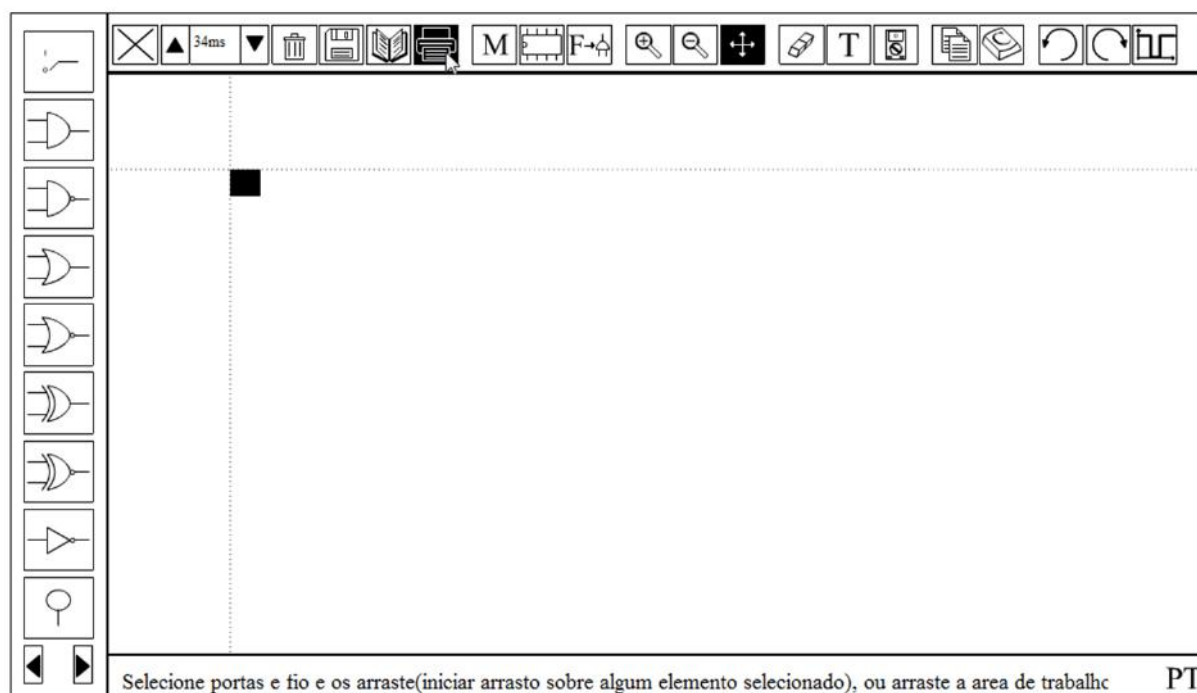


Figura 4.166 – Confirmação da impressão.

Após a janela de impressão do sistema operacional ser encerrada, o editor é encerrado, os marcadores e as linhas pontilhadas que delimitavam as regiões da Área de Trabalho são retiradas, os botões IMPRIMIR e INICIAR/CANCELAR voltam a exibir suas imagens originais.

Caso nenhum marcador de região tenha sido ativado, o processo de confirmação da impressão não é completado e é necessário ativar pelo menos um marcador para que o processo possa ser executado. Também é possível cancelar o editor selecionando o botão INICIAR/CANCELAR, o que faz com o editor seja encerrado, os marcadores e as linhas tracejadas que delimitavam as regiões sejam retirados da Área de Trabalho.

Para o caso da impressão dos circuitos internos de uma porta lógica de tipo Macros, através do Editor de Propriedades, existem algumas diferenças de operação no Editor de Impressão. A primeira diferença é que todas as operações descritas que utilizam o botão IMPRIMIR, nesse caso devem ser feitas com o botão de impressão do Editor de Propriedades. A segunda diferença é que a função de arrasto da Área de Trabalho é realizada pelos botões de navegação do Editor de Propriedades. A última diferença é que não é possível cancelar o editor, o que deve ser feito para encerrar o editor é confirmar a impressão com nenhum marcador selecionado.

4.6.7. Editor de Atalhos

Através do Editor de Atalhos é possível criar atalhos no teclado do computador para a adição rápida de portas lógicas na Área de Trabalho. Existem atalhos para todas as funções do programa e também é possível realizar a ligação entre as teclas de função do teclado e chaves lógicas na Área de Trabalho.

Para ativar o editor é necessário selecionar o botão ATALHOS e nesse momento a janela desse editor é inserida na Área de Trabalho. O botão GRÁFICOS passa a exibir uma imagem que informa que o editor está ativo.

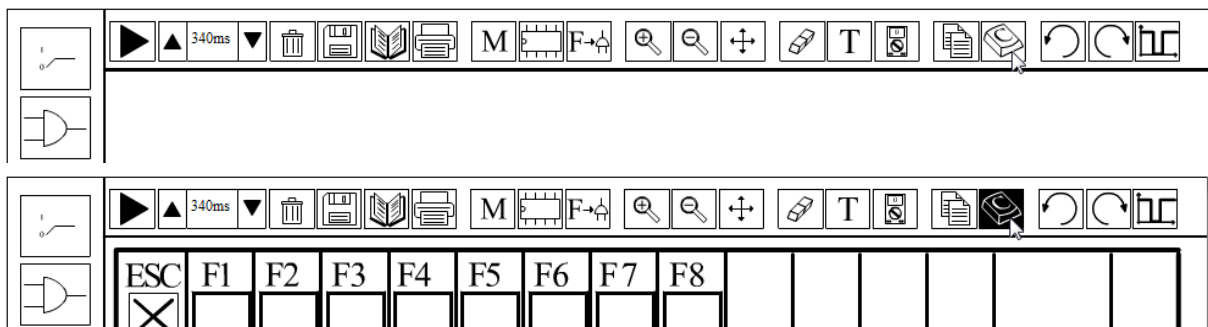


Figura 4.167 – Inicialização do Editor de Atalhos.

Como pode ser visto na Figura 4.168, a janela do editor apresenta uma imagem que representa um teclado, onde as teclas que têm letras ou números em seu interior são as teclas que interagem com esse editor. As teclas que possuem imagens dos botões da Caixa de Ferramentas em seu interior, ao serem pressionadas ativam a função referente ao botão em seu interior. Essas teclas não podem sofrer edição na função com a qual estão relacionadas, pois são padrão do programa. As teclas que possuem um quadrado de fundo branco em seu interior são as teclas que estão disponíveis para a criação de atalhos.

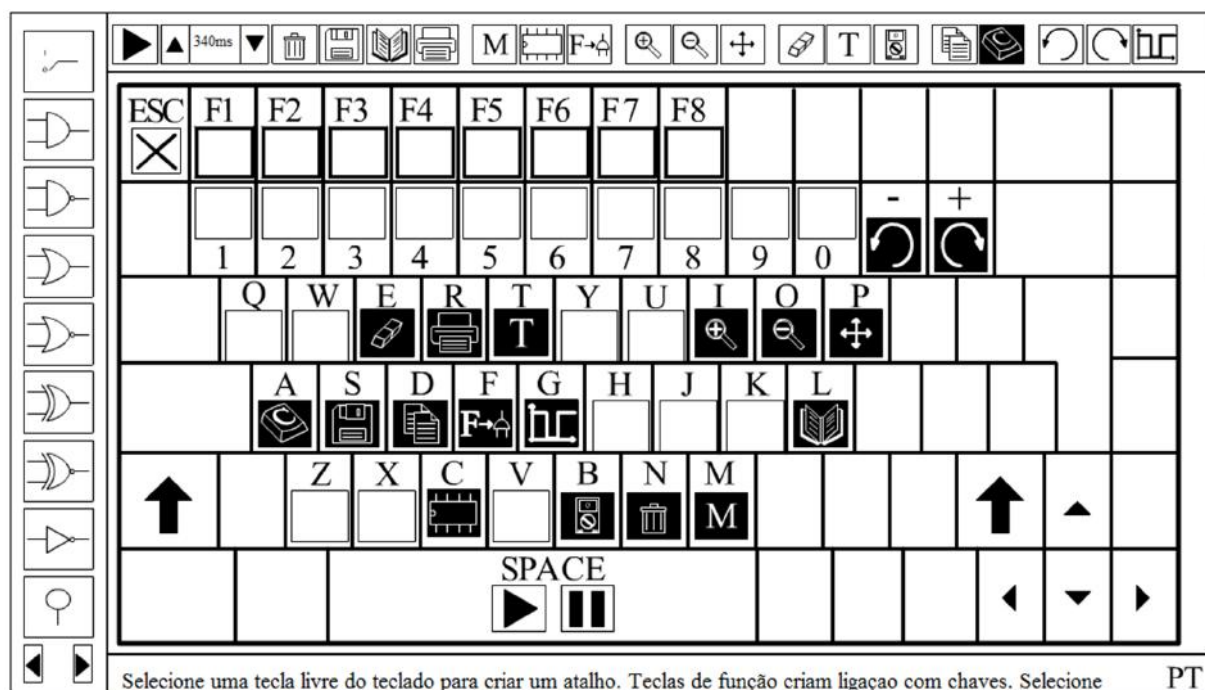


Figura 4.168 – Janela do Editor de Atalhos.

Excluindo-se as teclas de função, as demais teclas que possuem um quadrado de fundo branco em seu interior, podem ser utilizadas para a criação de atalhos de adição rápida de portas lógicas na Área de Trabalho. Para iniciar o processo de criação do atalho, deve-se posicionar o cursor do mouse sobre o quadrado de fundo branco da tecla em que o atalho será criado e pressionar o botão esquerdo do mouse. Nesse momento, o quadrado passa a exibir fundo preto, e deve-se procurar no Container de Portas, o ícone da porta lógica desejada, posicionar o cursor do mouse sobre esse ícone, pressionar e soltar o botão esquerdo do mouse. Assim o quadrado deixa de exibir o fundo preto e passa a exibir a imagem do ícone selecionado.

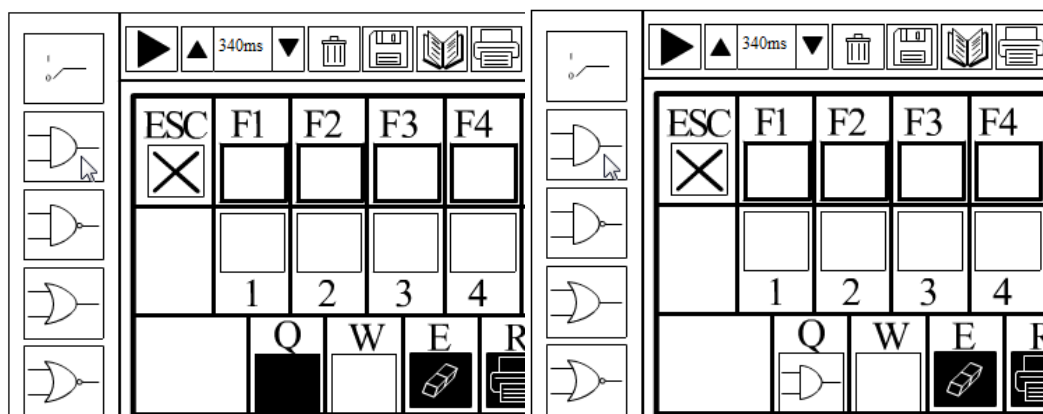


Figura 4.169 – Seleção do ícone para a criação do atalho para a tecla Q.

O processo ilustrado na Figura 4.169, pode ser repetido para todas as teclas disponíveis para a criação de atalho de adição rápida de portas lógicas na Área de Trabalho, não existindo nenhum tipo de restrição quanto ao tipo de porta lógica selecionada para o atalho. Caso seja necessário remover um atalho criado para uma tecla, deve-se posicionar o cursor do mouse sobre o ícone da porta lógica que encontra-se dentro da tecla na janela do editor e pressionar o botão esquerdo do mouse, o que fará com que o ícone da porta lógica seja substituído pelo quadrado de fundo preto. Nesse momento o atalho ainda não foi removido, apenas a tecla encontra-se selecionada, sendo necessário repetir o processo descrito, selecionando agora o quadrado de fundo preto, assim o atalho é removido e o quadrado de fundo branco passa a ser exibido na tecla que continha o ícone da porta lógica.

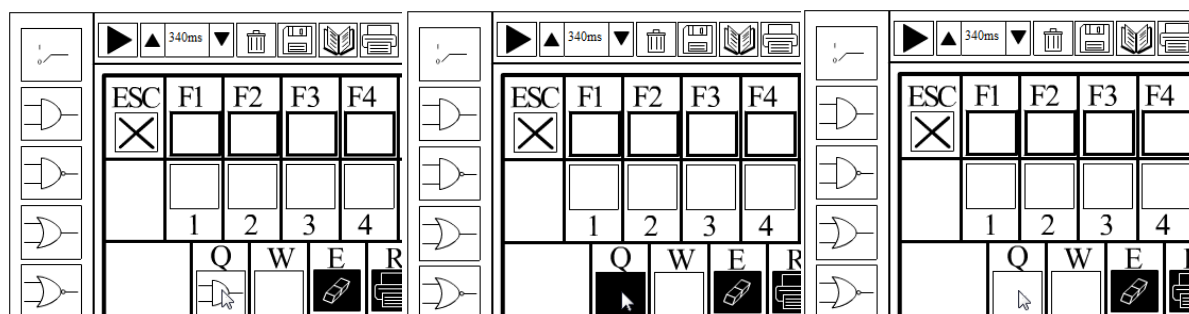


Figura 4.170 – Remoção de um atalho.

É possível realizar a substituição do atalho de uma tecla sem a necessidade de remover o atalho antigo e criar um novo. Deve-se iniciar o processo de remoção do atalho como ilustrado na Figura 4.169, porém não se deve selecionar o quadrado de fundo preto que é exibido na tecla, e sim selecionar no Container de Portas, o ícone da porta lógica para o novo atalho. Assim o ícone selecionado será exibido no lugar do quadrado de fundo preto e o novo atalho será criado, enquanto o antigo é automaticamente removido.

Após iniciar o processo de criação de um novo atalho ou o processo de remoção de um atalho criado, que resulta na exibição do quadrado de fundo preto na tecla em questão, é possível selecionar outra tecla qualquer, assim a tecla anterior deixa de exibir o quadrado de fundo preto e a nova tecla passa a exibi-lo. Esse procedimento pode ser considerado uma maneira de cancelar os processos de adição e remoção de atalhos.

Ao pressionar as teclas que possuem atalho de criação rápida de portas lógicas, o processo de adição de portas lógicas é iniciado, e também segue as mesmas regras detalhadas no capítulo 4.5.1. As demais teclas de atalho relacionadas às funções do programa, quando

pressionadas também executam suas respectivas funções, que também respeitam as regras descritas em seus respectivos capítulos.

As teclas de função possuem relação diferente das demais com o editor. Elas podem ser ligadas a chaves lógicas que estejam inseridas na Área de Trabalho, e assim quando uma dessas teclas é pressionada, o estado lógico da saída da chave lógica ligada a essa tecla é complementado. Assim é possível editar o estado lógico das saídas de varias chaves lógicas com maior agilidade. Para realizar a ligação entre uma tecla de função e uma chave lógica, deve-se posicionar o cursor do mouse sobre o quadrado de fundo branco que é exibido na tecla na janela do editor e pressionar o botão esquerdo do mouse. Nesse momento a janela do editor de atalhos é retirada da Área de Trabalho, e deve-se posicionar o cursor do mouse sobre uma chave lógica e pressionar o botão esquerdo do mouse, assim a janela do editor volta a ser exibida na Área de Trabalho, e o quadrado de fundo branco que foi selecionado passa a exibir fundo preto, indicando que a tecla que contém esse quadrado está ligada a uma chave lógica.

Na chave lógica onde a ligação com a tecla de função foi realizada, é inserida a imagem de um quadrado com um texto que informa qual é o nome da tecla de função que está ligada a chave. No caso da Figura 4.171, o quadrado inserido na chave exibe o nome "F1", que é o nome da tecla de função. Assim toda vez que a tecla de função F1 for pressionada e liberada, o estado lógico da saída dessa chave lógica será complementado, mesmo fora da execução da simulação.

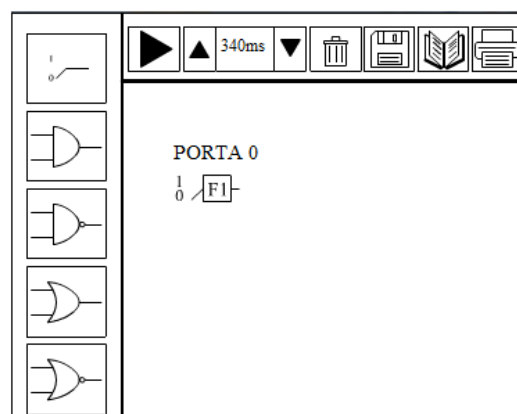


Figura 4.171 – Chave lógica com ligação com tecla de função.

Durante a seleção da chave lógica para realizar a ligação com uma tecla de função, é possível cancelar esse processo ao selecionar o botão, assim a janela do editor volta a ser exibida na Área de Trabalho e o processo de ligação é encerrado.

Para desfazer a ligação entre uma tecla de função e uma chave lógica, deve-se posicionar o cursor do mouse sobre o quadrado de fundo preto que é exibido na tecla de função na janela do editor de atalho e pressionar o botão direito do mouse. Nesse momento o quadrado da tecla passa a exibir fundo branco, indicando que não há ligação na tecla, e o quadrado com o nome da tecla de função é removido da chave lógica.

É possível realizar a mudança da chave lógica com a qual uma tecla de função tem ligação, sem a necessidade de remover a ligação anterior e criar a nova ligação com outra chave lógica. Para realizar o processo de mudança deve-se repetir o processo de criação da ligação, porém selecionando um quadrado de fundo preto de uma tecla de função que já possua ligação, assim é possível selecionar outra chave lógica e observar que depois da seleção, o quadrado que informa o nome da tecla de função deixa de ser exibido na chave lógica antiga e passa a ser exibido na chave lógica selecionada.

Para encerrar o editor deve-se selecionar o botão ATALHOS e nesse momento a janela do editor é retirada da Área de Trabalho e o botão ATALHOS volta a exibir sua imagem original. Qualquer processo que esteja sendo realizado pelo editor é terminado quando o editor é encerrado.

Capítulo 5

Conclusões

Neste trabalho foram apresentadas as bases teóricas necessárias ao entendimento da eletrônica digital. Apresentando a Álgebra de Boole, suas operações básicas e as portas lógicas que as representam. Foram apresentados também os circuitos integrados e foi detalhado um pouco sobre seu funcionamento.

Foi feito um estudo do comportamento de um circuito eletrônico digital, detalhando como os componentes que fazem parte desse circuito se comunicam e como as informações são transmitidas pelo circuito, apresentando as restrições nas montagens dos circuitos.

Foi apresentado e detalhado o modo como o funcionamento de um circuito eletrônico digital foi traduzido para um sistema virtual, juntamente com as considerações que foram necessárias para que a simulação pudesse corresponder ao desempenho de um circuito real. Foi apresentado o curto-circuito e como esse fenômeno foi abordado e traduzido no sistema virtual do programa.

A interface gráfica do programa foi inteiramente detalhada, assim como todas as funções e editores presentes no programa. Foram apresentados todos os procedimentos para inicializar e operar cada uma das funções e editores, informando as interações entre essas quando existiam. O Algoritmo de Quine - McCluskey foi apresentado, teve seus passos detalhados e estudados.

Alguns itens que foram programados no início do projeto do programa não puderam ser concluídos bem como problemas surgiram no funcionamento de algumas funções implementadas e fizeram com que elas tivessem de ser retiradas do programa final. Além desses, existem ainda algumas melhorias que podem ser feitas no programa, porém não houve tempo disponível para implementá-las. Esses itens foram deixados para trabalhos futuros que venham dar continuidade a este projeto:

- Finalização do gerenciador que procura por erros nos arquivos de dados carregados e adicionar a possibilidade de carregar arquivos de dados do programa Circuit Maker;
- Eliminação da dependência de que os fios tenham um número mínimo de quatro quinas;

- Adicionar mais opções de personalização no Editor de Impressão, como tamanho da região correspondendo a diferentes padrões de folhas não só o padrão A4, a possibilidade de imprimir em uma só folha qualquer seção da Área de Trabalho;
- Criar uma função que permita realizar a secção de um fio e outra que permita realizar a união entre dois fios;
- Implementação de um editor para criação de máquinas sequenciais e contadores;
- Criar a função "Desfazer" para que se possa desfazer uma operação realizada;
- Criar mais opções no Editor de Interpretação de Funções para restringir o tipo das portas lógicas que podem ser utilizadas;
- Adicionar no Editor de Minimização outros métodos de resolução além do algoritmo de Quine - McCluskey;
- Criar um editor para inserção de caixas de texto informativas na área de trabalho;
- Terminar o desenvolvimento dos modelos dos circuitos integrados 7490, 7493, 74175 e 74181;
- Melhorar os métodos de exibição e impressão dos gráficos do Editor de Gráficos;
- Revisão completa do design da interface gráfica;

O programa desenvolvido passou por diversas atualizações antes da versão final e durante esse processo suas funções e editores foram aperfeiçoados e os problemas encontrados resolvidos. O programa permite ao usuário acesso rápido a todas as suas funções, e cobre tópicos como projeto e simulação de circuitos eletrônicos digitais, minimização de funções e criação de circuitos a partir de uma função booleana. Este ainda possui funções de carregamento e salvamento de arquivo de dados, para que seja possível continuar a trabalhar num mesmo circuito, em computadores diferentes. Além dessa função existe a função de impressão para que se possa analisar o circuito fora do programa. O editor de atalhos, além de permitir a criação de atalhos para inserção rápida de portas lógicas, permite ao usuário fazer ligações entre as teclas de função do teclado e chaves lógicas do circuito, podendo modificar seus estados lógicos através dessas teclas em tempo real durante a simulação. O editor de gráficos permite ao usuário um meio de análise do comportamento do mesmo, posterior à simulação, e também há possibilidade de impressão dos gráficos criados. O Editor de Macros permite criar os circuitos internos de circuitos integrados e inserí-los na Área de Trabalho.

O programa foi implementado como planejado e encontra-se em uma versão estável e pronta para a utilização.

Capítulo 6

Bibliografia

- [1] SOUZA, Antonio Lopes. de, HAZAN, Sergio Sami. “Circuitos Lógicos”, Departamento de Engenharia Elétrica, Universidade Federal do Rio de Janeiro.
- [2] ARAÚJO, Lucínio Preza de, “Circuitos Integrados”, Escola Secundária Carlos Amarante, Braga, Portugal. Disponível em: <http://www.prof2000.pt/users/lpa/circuitos%20integrados.ppt>
- [3] <http://www.datasheetarchive.com/>
- [4] MURRELL, Stephen. "74-series IC data sheet index". Electrical and Computer Engineering, College of Engineering, University of Miami. Disponível em: <http://rabbit.eng.miami.edu/info/datasheets/>
- [5] On Semiconductor. “LS TTL Data DL121/D”. 6ª edição, Colorado, Literature Distribution Center for ON Semiconductor, 2000. Disponível em: <http://www.ecse.rpi.edu/courses/CStudio/data%20sheets/DL121-D.PDF>
- [6] CARDOSO, João Manuel Paiva. “*Método de Quine-McCluskey*”, Department of Electronics and Informatics Engineering (DEEI), Faculty of Sciences and Technology, University of Algarve. Disponível em: <http://w3.ualg.pt/~jmcardo/ensino/sistdigitais01/AulasTeoricas%5Caula12.pdf>
- [7] LACERDA, Willian Lacerda. “*Projeto e Implementação de Circuitos Classificadores Digitais com Controle da Generalização Baseado na Regra do Vizinho-mais-próximo Modificada*”. Tese de doutorado. Programa de Pós-Graduação em Engenharia Elétrica. Universidade Federal de Minas Gerais, Minas Gerais, Fevereiro de 2006. Disponível em: http://www.bibliotecadigital.ufmg.br/dspace/bitstream/handle/1843/BUOS-8CTFF8/wilian_soares_lacerda.pdf
- [8] COSTA, Miguel Antonio Bueno de. “*Simulações de Sistemas*”. Departamento de Engenharia de Produção. Universidade Federal de São Carlos. Disponível em: http://www.simucad.dep.ufscar.br/dn_sim_doc01.pdf
- [9] CASTRO, Eberval Oliveira. “*Introdução ao uso do simulador de circuitos CircuitMaker Student 6.2c para o apoio no ensino-aprendizado de Eletricidade e Eletrônica*”. Faculdade Anhanguera de Jundiaí. Disponível em: <ftp://ftp.sm.ifes.edu.br/professores/fbento/sisdig/CircuitMaker/Tutorial-Circuit-Maker-Prof-eberval-v2.pdf>

ANEXO 1

Neste anexo serão apresentados todos os passos e procedimentos que compõem o Algoritmo de Quine-McCluskey [6] e [7]. Inicialmente será feita a introdução ao problema de se criar e simplificar uma função booleana.

Uma função booleana pode ser definida como uma função que apresenta como resposta somente estado lógico zero ou estado lógico um, e em que essa resposta depende unicamente dos estados lógicos das variáveis de entrada, que também só podem assumir valores zero ou um. Assim, diferentes combinações de estados lógicos das variáveis de entrada resultam em diferentes resultados para a função. Uma maneira de enumerar todas essas combinações de estados das variáveis de entrada e a resposta da função a esses estados, é a criação da Tabela Verdade dessa função. Nessa tabela devem-se inserir todas as combinações de estados das variáveis de entrada, e para cada combinação inserir a resposta da função.

A	B	C	f(A,B,C)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Figura A.1 – Tabela Verdade de uma função de três variáveis de entrada.

Na Figura A.1 a Tabela Verdade da função "f" exibe os diferentes estados de resposta dessa função. Após criar a Tabela Verdade, o passo seguinte é a obtenção de uma representação algébrica para esta, que deve ser constituída por um conjunto de operações booleanas que envolvam as variáveis de entrada para definir a resposta da função "f". Porém esse não é um processo trivial uma vez que a quantidade de combinações das variáveis de entrada pode representar um problema para a identificação dessas operações.

Existe um processo que facilita a obtenção da representação algébrica da Tabela Verdade. Esse processo tem duas formas de ação, uma que cria a representação algébrica da tabela verdade tomando como referência apenas as combinações das variáveis de entrada que impliquem em estado lógico um na saída da função, e outra que toma como referências as combinações das variáveis de entrada que impliquem em estado lógico zero na saída da função. A primeira forma de ação resulta em uma representação algébrica constituída apenas

pela soma de termos, onde cada um desses termos é um produto das variáveis de entrada, e a segunda forma de ação resulta em uma representação algébrica constituída apenas pelo produto de termos, onde cada um desses termos é uma soma das variáveis de entrada.

Na primeira forma de ação, cada um dos termos da soma é chamado de Mintermo, e este pode ser definido como sendo o produto lógico de todas as variáveis de entrada da função, e nesse produto essas variáveis podem ou não estar negadas [1]. Cada combinação de estados lógicos das variáveis de entrada possui um Mintermo associado e este é construído de maneira que quando essa combinação é aplicada no produto que representa o Mintermo, este sempre leva a um resultado igual a um. Uma maneira simples de satisfazer essa condição é observar a combinação das variáveis de entrada e realizar o seguinte passo: caso a variável apresente estado lógico zero na combinação, ela deve ser negada no produto, e caso apresente estado lógico um na combinação, ela não deve ser negada no produto. Assim o produto criado só leva ao estado lógico um quando é aplicada a ele a combinação das variáveis de entrada que foi usada como referencia na sua criação. Os Mintermos que devem fazer parte da soma devem ser aqueles que estão associados à combinação de variáveis de entrada que levem o resultado da função ao estado lógico um. Esses Mintermos foram selecionados na Figura A.2 com um “*”.

A	B	C	f(A,B,C)	Mintermo
0	0	0	0	$\overline{A}.\overline{B}.\overline{C}$
0	0	1	0	$\overline{A}.\overline{B}.C$
0	1	0	1	$\overline{A}.B.\overline{C}$ *
0	1	1	1	$\overline{A}.B.C$ *
1	0	0	1	$A.\overline{B}.\overline{C}$ *
1	0	1	1	$A.\overline{B}.C$ *
1	1	0	0	$A.B.\overline{C}$
1	1	1	1	$A.B.C$ *

Figura A.2 – Enumeração dos Mintermos e seleção dos Mintermos que farão parte da função.

No caso da segunda forma de ação, cada um dos termos do produto é chamado de Maxitermo, e este pode ser definido como sendo a soma lógica de todas as variáveis de entrada da função, e nessa soma as variáveis podem ou não ser negadas [1]. Cada combinação de estados lógicos das variáveis de entrada possui um Maxitermo associado e sua construção é semelhante à construção dos Mintermos, porém a soma que representa o Maxitermo deve levar a um resultado igual a zero quando a combinação de variáveis de entrada associadas a esse Maxitermo for aplicada à soma. Assim, a construção do Maxitermo deve seguir o seguinte passo: caso a variável apresente estado lógico um na combinação, ela deve ser

negada na soma, e caso apresente estado lógico zero na combinação, ela não deve ser negada na soma. Assim, a soma construída só levará a um resultado igual a zero quando a combinação referente ao Maxitermo construído for aplicada nessa soma. Os Maxitermos que devem fazer parte do produto devem ser aqueles que estão associados à combinação de variáveis de entrada que levem o resultado da função ao estado lógico zero. Esses Maxitermos foram selecionados na Figura A.3 com um “*”.

A	B	C	f(A,B,C)	Maxitermo
0	0	0	0	$A+B+C$ *
0	0	1	0	$A+B+\overline{C}$ *
0	1	0	1	$A+\overline{B}+C$
0	1	1	1	$A+\overline{B}+\overline{C}$
1	0	0	1	$\overline{A}+B+C$
1	0	1	1	$\overline{A}+B+\overline{C}$
1	1	0	0	$\overline{A}+\overline{B}+C$ *
1	1	1	1	$\overline{A}+\overline{B}+\overline{C}$

Figura A.3 – Enumeração dos Maxitermos e seleção dos Maxitermos que farão parte da função.

Assim, após selecionar os Mintermos da função e organizá-los em uma soma, está formada a representação algébrica da tabela verdade da função. Essa representação algébrica também pode ser representada pelo produto de todos os Maxitermos selecionados. A Figura A.4 exibe as duas representações algébricas criadas para a função. Ambas apresentam o mesmo resultado para uma mesma combinação de variáveis de entrada. A diferença na quantidade e tipo de operações é resultado dos diferentes métodos utilizados para construí-las.

$$f(A,B,C) = \overline{A}.B.\overline{C} + \overline{A}.B.C + A.\overline{B}.\overline{C} + A.\overline{B}.C + A.B.C$$

$$f(A,B,C) = (A+B+C).(A+B+\overline{C}).(\overline{A}+\overline{B}+C)$$

Figura A.4 – Duas representações algébricas para uma mesma função.

A representação algébrica da função por soma de Mintermos apresentada na Figura A.4 possui um número grande de termos e a implementação do Circuito Lógico associado a essa função exigirá a utilização de um grande número de portas lógicas. Assim caso seja possível simplificar a estrutura da função, apresentando uma função com menor número de termos e em que cada termo tenha menos variáveis de entrada envolvidas, o Circuito Lógico referente a essa função também será simplificado.

Existem alguns métodos para realizar a simplificação das funções booleanas e entre eles está o método que utiliza o Algoritmo de Quine - MacCluskey. Esse método é aplicado sobre a função em sua forma canônica de soma de produtos, ou seja, ele utiliza os Mintermos da

função para a obtenção de uma representação algébrica simplificada da função, e essa nova representação deixa de ser considerada uma forma canônica. O Algoritmo de Quine-McCluskey é um método projetado para encontrar os Primos Implicantes de uma função booleana e realizar a seleção do conjunto de Primos Implicantes que cubram toda a função [9].

A ideia por trás desse método é de aplicar sucessivamente o Teorema da Álgebra de Boole que descreve a Propriedade de Absorção, ilustrada na Figura A.5, sobre os Mintermos da função [1]. A aplicação dessa propriedade acaba por eliminar dois termos e formar um terceiro com um número de variáveis menor que os anteriores.

$$\begin{aligned}\overline{A}.B + A.B &= B.(\overline{A} + A) \\ &= B.(1) \\ &= B \\ \overline{A}.B + A.B &= B\end{aligned}$$

Figura A.5 – Propriedade de absorção aplicada a uma função lógica.

O termo que resulta da aplicação desse processo recebe o nome de Implicante. Esse processo é aplicado sobre os Mintermos com a finalidade de encontrar todas as possibilidades de simplificações destes, sendo possível que um mesmo Mintermo seja utilizado em dois ou mais processos. Esse processo pode ser aplicado também entre os Implicantes criados, para a obtenção de Implicantes com menor número de variáveis. A Figura A.6 ilustra a aplicação desse processo sobre os Mintermos de uma função para simplificar sua representação algébrica.

$$\begin{aligned}f(A,B,C) &= \overline{A}.B.\overline{C} + \overline{A}.B.C + A.\overline{B}.\overline{C} + A.\overline{B}.C + A.B.C \\ &= \overline{A}.B + A.\overline{B} + A.B.C\end{aligned}$$

Figura A.6 – Simplificação de uma função lógica.

A aplicação do processo na função da Figura A.6 resultou na eliminação de quatro Mintermos e na criação de dois Implicantes. Porém a simplificação realizada não é a única possível, e a Figura A.7 ilustra outra simplificação que pode ser feita sobre a função.

$$\begin{aligned}f(A,B,C) &= \overline{A}.B.\overline{C} + \overline{A}.B.C + A.\overline{B}.\overline{C} + A.\overline{B}.C + A.B.C \\ &= \overline{A}.B + A.\overline{B}.\overline{C} + A.C\end{aligned}$$

Figura A.7 – Simplificação de uma função lógica.

Analisando-se as Figuras A.6 e A.7 é possível observar que a função resultante da aplicação do processo que utiliza a propriedade de absorção, pode apresentar diferentes configurações e o nível de simplificação alcançado por cada uma dessas configurações depende da habilidade de se identificar pares de Mintermos em que a propriedade de absorção possa ser aplicada.

O processo descrito pode ser bastante complexo de ser aplicado para funções com grande número de variáveis de entrada, pois a identificação de quais pares de Mintermos satisfazem a propriedade de absorção é um processo que demanda bastante tempo e análise, e a grande quantidade de configurações diferentes que podem ser obtidas para a função, faz com que a identificação da configuração dentre estas, que apresenta o maior nível de simplificação torne-se um trabalho muito longo. Assim o Algoritmo de Quine - MacCluskey apresenta um método para identificar os primos implicantes com maior eficiência e para a escolha dos Primos Implicantes e Mintermos que farão parte da função.

O algoritmo será aplicado para uma nova função que possui quatro variáveis de entrada, com um grande número de Mintermos na soma, e como foi descrito representaria um grande esforço para ser simplificada sem a utilização do algoritmo. A Figura A.8 apresenta a função e a soma dos seus Mintermos.

$$f(A,B,C,D) = \overline{A}.\overline{B}.\overline{C}.\overline{D} + \overline{A}.\overline{B}.\overline{C}.D + \overline{A}.\overline{B}.C.\overline{D} + \overline{A}.\overline{B}.C.D + \overline{A}.B.\overline{C}.\overline{D} + \overline{A}.B.\overline{C}.D + \overline{A}.B.C.\overline{D} + \overline{A}.B.C.D + A.\overline{B}.\overline{C}.\overline{D} + A.\overline{B}.\overline{C}.D + A.\overline{B}.C.\overline{D} + A.\overline{B}.C.D + A.B.\overline{C}.\overline{D} + A.B.\overline{C}.D + A.B.C.\overline{D} + A.B.C.D$$

Figura A.8 – Função lógica representada como soma de Mintermos.

Os passos do algoritmo são os seguintes:

1. Agrupar os Mintermos da função pelo número de variáveis não negadas que cada um deles possui;

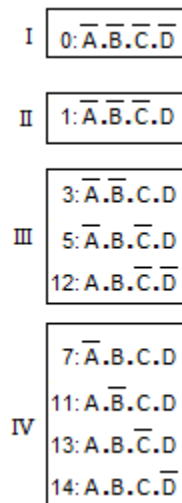


Figura A.9 – Grupos de Mintermos com mesmo número de variáveis não negadas.

2. Realizar inspeção, cruzando cada grupo com o grupo imediatamente abaixo deste;
 - 2.1. A cada cruzamento devem ser analisados pares de Mintermos em busca de pares em que os produtos que representam os Mintermos são idênticos, a não ser por uma variável que deve estar complementada em um produto e não complementada no outro produto;
 - 2.2. Os pares devem ser compostos sempre por Mintermos de grupos diferentes. Os pares que satisfizerem a condição, dentro do cruzamento dos grupos, devem ser agrupados em um novo grupo, e neste grupo deve ser exibido o Implicante resultante da aplicação da propriedade de absorção sobre o par de Mintermos.
 - 2.3. Cada Mintermo utilizado durante a análise para formar um Implicante é marcado com um asterisco;

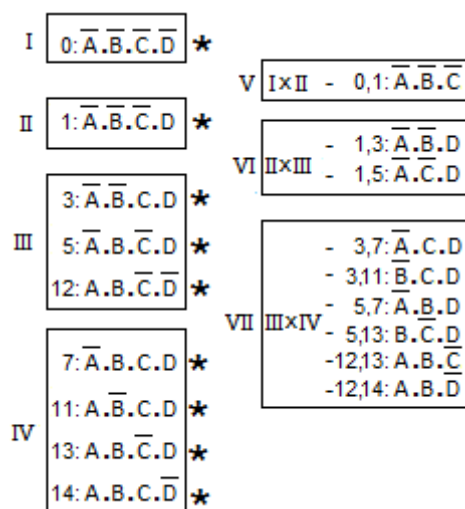


Figura A.10 – Grupos V, VI e VII criados.

3. Refazer a inspeção descrita no segundo item sobre os novos grupos;

- 3.1. A mesma inspeção do segundo item é refeita cruzando somente os novos grupos que foram criados no último passo, porém neste caso as análises devem ser feitas somente entre os Implicantes que possuem as mesmas variáveis.
- 3.2. Repetir item 3.1 até que o cruzamento não resulte em novos grupos.

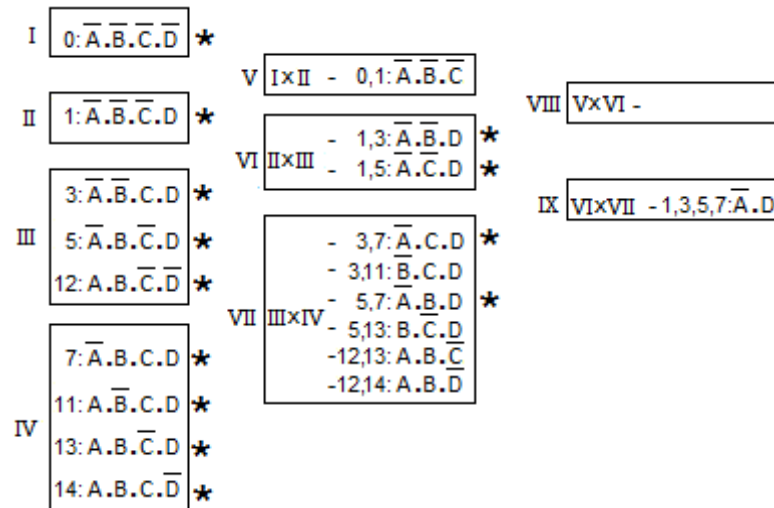


Figura A.11 – Finalizada a criação dos grupos, restam os grupos I, II, III, IV, V, VI, VII, e IX.

4. Listar os Mintermos e os Implicantes que não participaram da formação de Implicantes;
- 4.1. Os termos que não estiverem marcados por um asterisco são selecionados, e os Implicantes que não participaram da formação de novos Implicantes são chamados de Primos Implicantes;

0,1: $\overline{A}.\overline{B}.\overline{C}$
 3,11: $\overline{B}.C.D$
 5,13: $B.\overline{C}.D$
 12,13: $A.B.\overline{C}$
 12,14: $A.B.\overline{D}$
 1,3,5,7: $A.D$

Figura A.12 – Relação dos Primos Implicantes da função.

5. Criar tabela;
- 5.1. Inserir nas células da primeira linha todos os Mintermos da função;
- 5.2. Inserir nas células da primeira coluna todos os Primos Implicantes e os Mintermos listados no item 4;
- 5.3. Preencher as demais células da tabela com um "X" quando a linha que a define represente um Primo Implicante que contenha o Mintermo referente à coluna que a define;
- 5.4. Manter vazia qualquer célula que não satisfizer a condição descrita no item 5.3;

		MINTERMO									
		0	1	3	5	7	11	12	13	14	
I M P L I C A N T E	0,1	X	X								
	3,11			X			X				
	5,13				X				X		
	12,13							X	X		
	12,14							X		X	
	1,3,5,7		X	X	X	X					

Figura A.13 – Tabela para escolha dos Primos Implicantes que cobrem toda a função.

6. Determinar Primos Implicantes essenciais;

6.1. Determinar através da tabela criada no item 5, os Primos Implicantes essenciais, os demais Primos Implicantes e os Mintermos necessários para cobrir toda a função;

6.2. Construir a nova representação algébrica da função como o somatório de todos os Primos Implicantes e Mintermos selecionados;

A determinação dos Primos Implicantes essenciais é feita pela análise da tabela criada no quinto item, observando quais as colunas possuem apenas uma célula preenchida por um "X". Assim, o Primo Implicante correspondente à linha que define essa célula é um Primo Implicante essencial. A Figura A.14 ilustra essa seleção.

		MINTERMO								
		0	1	3	5	7	11	12	13	14
I M P L I C A N T E	0,1	X	X							
	3,11			X			X			
	5,13				X				X	
	12,13							X	X	
	12,14							X		X
	1,3,5,7		X	X	X	X				

ESSENCIAIS

0,1
3,11
12,14
1,3,5,7

ESSENCIAIS

0,1
3,11
12,14
1,3,5,7

Figura A.14 – Definição dos Primos Implicantes essenciais.

Os Mintermos que forem cobertos pelos Primos Implicantes essenciais podem ser desconsiderados nas análises seguintes realizadas na tabela através da marcação com um asterisco na célula deste Mintermo. A escolha dos demais Primos Implicantes deve ser feita para cobrir todos os Mintermos restantes da função, aqueles cujas células não possuem um asterisco. Não há nenhuma regra específica para tal seleção, porém é aconselhável selecionar o menor número possível de Primos Implicantes que cubram toda a função. No caso da Figura A.15 apenas o Mintermo “13” não está marcado e o Primo implicante “5,13” foi escolhido para cobri-lo.

		MINTERMO									
		0*	1*	3*	5*	7*	11*	12*	13	14*	
I M P L I C A N T E	0,1 *	X	X								
	3,11 *			X			X				
	5,13				X				X		
	12,13							X	X		
	12,14 *							X		X	
	1,3,5,7 *		X	X	X	X					

5,13

Figura A.15 – Escolha dos Primos Implicantes não essenciais.

A nova representação algébrica da função assim como foi descrito no sexto item é construída como a soma dos Primos Implicantes essenciais selecionados na Figura A.14 e o Primo Implicante não essencial selecionado na Figura A.15.

$$f(A,B,C,D) = \overline{A}.\overline{B}.\overline{C} + \overline{B}.C.D + B.\overline{C}.D + A.B.\overline{D} + \overline{A}.D$$

Figura A.16 – Nova representação algébrica da função.

No caso de existir na Tabela Verdade alguma célula com valor "don't care", o Mintermo referente a essa célula deve ser considerado para o primeiro item do método, porém não deve ser incluído na tabela criada no quinto item.

Durante o processo de implementação desse método no Editor de Minimização, houve a adição de um processo de seleção dos Primos Implicantes não essenciais, focado em reduzir o número de Primos Implicantes selecionados. Esse processo realiza a checagem dos Primos Implicantes e identifica quando um Primo Implicante cobre os mesmos Mintermos de um segundo Primo Implicante, e elimina esse segundo Primo Implicante da tabela. Em seguida os Primos Implicantes restantes são analisados e o menor número destes é selecionado mantendo a cobertura total da função. Esse processo geralmente acaba resultando na seleção de Primos Implicantes cujos termos possuem poucas variáveis.

APÊNDICE A

